

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки

Кафедра технічної кібернетики

«На правах рукопису»  
УДК 004.043

«До захисту допущено»

Завідувач кафедри  
\_\_\_\_\_ І.Р. Пархомей  
(підпис)

“ ” \_\_\_\_\_ 2018 р.

**Магістерська дисертація**

**на здобуття ступеня магістра**

зі спеціальності 121 «Інженерія програмного забезпечення»

на тему: Система виділення динамічних об'єктів на серії зображень за допомогою нейронної мережі

Виконав: студент другого курсу, групи ІТ-74мп  
(шифр групи)

\_\_\_\_\_ Плохута Дмитро Олександрович \_\_\_\_\_  
(прізвище, ім'я, по батькові) (підпис)

Науковий керівник доцент, к.т.н., доц. Корнага Я.І. \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант \_\_\_\_\_  
(назва розділу) (науковий ступінь, вчене звання, прізвище, ініціали) (підпис)

Рецензент \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.

Студент \_\_\_\_\_  
(підпис)

Київ – 2018 року

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки

Кафедра технічної кібернетики

Рівень вищої освіти – другий (магістерський)

Спеціальність 121 «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри

\_\_\_\_\_ І.Р. Пархомей  
(підпис)

«\_\_\_» \_\_\_\_\_ 2018 р.

**ЗАВДАННЯ**  
**на магістерську дисертацію студенту**  
**Плохуті Дмитру Олександровичу**  
(прізвище, ім'я, по батькові)

1. Тема дисертації «Система виділення динамічних об'єктів на серії зображень за допомогою нейронної мережі», \_\_\_\_\_  
науковий керівник дисертації Корнага Ярослав Ігорович, к.т.н., доц.  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «\_\_\_» \_\_\_\_\_ 2018 р. № \_\_\_\_\_

2. Термін подання студентом дисертації \_\_\_\_\_

3. Об'єкт дослідження – процес автоматизованого управління обробки зображень.

4. Предмет дослідження – моделі та методи інформаційних технологій для обробки зображень.

5. Перелік завдань, які потрібно розробити – аналіз проблеми та існуючих рішень; аналіз і реалізація методу; розробка додатку; дослідження ефективності розробленого додатку.

6. Орієнтовний перелік ілюстративного матеріалу – три плакати та три креслення

7. Орієнтовний перелік публікацій – одна публікація

## 8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

9. Дата видачі завдання \_\_\_\_\_

## Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Аналіз предметної області	13.09.2018 р.	
2	Постановка задачі	15.09.2018 р.	
3	Аналіз інформаційного забезпечення	20.09.2018 р.	
5	Аналіз алгоритмічного забезпечення	25.09.2018 р.	
6	Розробка алгоритмічного забезпечення	15.10.2018 р.	
7	Розробка програмного забезпечення	01.11.2018 р.	
8	Маркетинговий аналіз стартап-проекту	10.11.2018 р.	
9	Висновки	15.11.2018 р.	

Студент

\_\_\_\_\_  
(підпис)

Д.О. Плохута  
(ініціали, прізвище)

Науковий керівник дисертації

\_\_\_\_\_  
(підпис)

Я.І. Корнага  
(ініціали, прізвище)

## АНОТАЦІЯ

У роботі розглянуто проблему в області автоматизованого редагування фотографій, показано основні особливості існуючих рішень та додатків, їх переваги та недоліки.

При фотографуванні різних краєвидів, будинків та інших нерухомих об'єктів, на фотографії попадають багато рухомих об'єктів, щоб їх прибрати треба використовувати різні програми. У них є багато можливостей для редагування зображень, проте їх не вистачає для повноцінної роботи з зображеннями.

Визначено завдання для системи виділення динамічних об'єктів на серії зображень за допомогою нейронної мережі, відібрано нейронну мережу та спосіб її навчання, який найбільш підходить для даної задачі. Описано структуру нейронної мережі та проведено експерименти по навчанню та її роботі.

Ця система надає вам автоматизований пошук та видалення рухомих об'єктів без використання важких для навчання додатків та програм на персональні комп'ютери. Дозволяє зменшити витрати часу та повністю забезпечити видалення рухомих об'єктів з фотографій.

Ключові слова: нейронна мережа, редагування зображень, навчання, розпізнавання, фотографія.

Розмір пояснювальної записки – 92 аркуша, містить 30 ілюстрацій, 23 таблиць, 6 додатків.

## ABSTRACT

The paper considers the problem in the field of automated photo editing, shows the main features of existing solutions and applications, their advantages and disadvantages.

When photographing different landscapes, buildings and other immovable objects, many moving objects fall into the photo, to remove them, you need to use different programs. They have many features for editing images, but they are not enough for full-fledged image manipulation.

The tasks for the system of search of dynamic objects on a series of images with a help of the neural network are determined, and the neural network and the method of training are selected which are most suitable for this task. The structure of the neural network is described and experiments on training and work.

This system provides you with automated search and removal of moving objects without the use of hard-to-learn apps and programs on personal computers. Allows you to save time and completely remove moving objects from your photos.

Keywords: neural network, image editing, learning, recognition, photography.

Explanatory note size – 92 pages, contains 30 illustrations, 23 tables, 6 applications.

**Пояснювальна записка  
до магістерської дисертації**

на тему: Система виділення динамічних об'єктів на серії зображень  
за допомогою нейронної мережі

Київ – 2018 року

## ЗМІСТ

ВСТУП.....	11
Розділ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАДАЧІ.....	13
1.1. Огляд існуючих рішень .....	13
1.1.1. Remove.....	13
1.1.2. Adobe Photoshop Fix .....	14
1.1.3. Zoner Photo Studio.....	15
1.2. Постановка Задачі .....	16
1.2.1. Об'єкт та предмет дослідження.....	16
1.2.2. Переваги системи виділення динамічних об'єктів на серії зображень за допомогою нейронної мережі .....	17
1.2.3. Недоліками системи виділення динамічних об'єктів на серії зображень за допомогою нейронної мережі.....	17
Висновки по розділу .....	18
Розділ 2. Аналіз відомих нейронних мережі та вибір інструментів розробки	19
2.1. Штучна нейронна мережа .....	19
2.1.1. Історія виникнення нейронної мережі.....	19
2.1.2. Застосування нейронних мереж до різних сферах .....	20
2.1.3. Нейронна мережа в роботі.....	21
2.1.4. Методи навчання нейронної мережі.....	23
2.2. Згортова нейронна мережа .....	24
2.2.1. Основна структура.....	27
2.2.2. Додаткові шари рекурентної нейронні мережі.....	30
2.2.3. Регуляризація та збільшення даних.....	31
2.2.4. Виявлення згорткового об'єкта .....	32
2.2.5. Продуктивність R-CNN.....	34
2.2.6. Створення та використання пропозиції.....	35
2.2.7. Selective Search .....	37

2.2.8. Edge Boxes .....	38
2.3. Автокодувальник.....	38
2.4. Рекурентні нейронні мережі.....	40
2.5. Обґрунтування вибору програмних засобів .....	40
2.5.1. Java .....	41
2.5.2. JSON.....	42
2.5.3. Kotlin .....	42
2.5.4. Phyton .....	43
2.5.5. Camera API .....	43
2.6. Розробка Android додатку .....	44
2.6.1. Інструменти розробки .....	44
2.6.2. Програмні пакети .....	46
Висновок по розділу.....	46
Розділ 3. РОЗРОБКА АЛГОРИТМІЧНОГО ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ .....	48
3.1. Експеримент згорткової нейронної мережі.....	48
3.1.1. Пошук об'єкта .....	49
3.1.2. Об'єкти у перспективі .....	52
3.2. Набір даних .....	52
3.2.1. Перевірка даних.....	53
3.3. Реалізація.....	55
3.3.1. Використане обладнання.....	55
3.3.2. MatConvNet та швидкий R-CNN .....	56
3.3.3. Геометричний висновок .....	57
3.4. Оцінка.....	58
3.4.1. Швидкий R-CNN .....	58
3.4.2. Позитивні помилки та негативні помилки .....	60
3.4.3. Не максимальна придушення .....	62



3.4.4. Геометричний висновок .....	63
3.5. Android додаток.....	66
3.5.1. Клієнтський інтерфейс Android додатку .....	66
3.5.2. Структура роботи андроїд додатку .....	69
3.5.3. Структура роботи серверної частини .....	71
Висновки по розділу .....	72
Розділ 4. МАРКЕТИНГОВИЙ АНАЛІЗ СТАРТАП-ПРОЕКТУ .....	74
4.1. Опис ідеї проекту .....	74
4.2. Технологічний аудит ідеї проекту.....	75
4.3. Аналіз ринкових можливостей запуску стартап-проекту.....	75
4.4. Розроблення ринкової стратегії проекту .....	83
4.5. Розроблення маркетингової програми стартап-проекту .....	85
Висновки по розділу .....	88
ВИСНОВКИ.....	89
ПЕРЕЛІК ПОСИЛАНЬ .....	92
ДОДАТКИ.....	94

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

БД – База даних

СУБД – Система управління базами даних

HTML – HyperText Markup Language

ПЗ – Програмне забезпечення

API – Application programming interface

SQL – Structured Query Language

UML – Unified Modeling Language

CPU – Central Processing Unit

FC – Fully Connected

GPU – Graphics Processing Unit

RoI – Агрегування областей інтересу

## ВСТУП

В сучасному світі технології швидко розвиваються, і в кожному гаджеті є вбудована камера та навіть не дорогі пристрої мають досить хорошу камеру, і ці технології можна використовувати в багатьох сферах.

Оцінки Infotrends показують, що в 2016 році фотоапарати та мобільні пристрої зафіксували більше 1,1 трильйона зображень. Згідно з тією самою оцінкою, в 2020 році зросте до 1,4 трлн. Багато хто з цих зображень зберігаються в хмарних сервісах або публікуються в інтернеті. У 2014 році понад 1,8 мільярда зображень завантажувались щодня на найпопулярніші платформи, такі як Instagram та Facebook.

Виходячи за межі споживчих пристроїв, у всьому світі є камери, які знімають зображення для автоматизації. Автомобілі відстежують дорогу, і камери контролюють ті самі машини. Роботам потрібно розуміти візуальну сцену, щоб збирати пристрої та сортувати відходи. Вони використовуються інженерами, лікарями та космічними дослідниками.

Протягом останнього десятиліття в області машинного навчання переважають так звані глибокі нейронні мережі, які користуються перевагами підвищення обчислювальної потужності та доступності даних. Підтип нейронної мережі під назвою збірної нейронної мережі (CNN) добре підходить для завдань, пов'язаних із зображеннями. Мережа навчається шукати різні функції, такі як краї, кути та кольорові відмінності, по всьому зображенню та об'єднати їх у більш складні форми. Для виявлення об'єкта система повинна як оцінити розташування можливих об'єктів, так і класифікувати їх.

При фотографуванні різних краєвидів, будинків і т.д., на фотографії попадають багато невідомих людей, щоб їх прибрати треба використовувати різні програми. У них є багато можливостей для редагування зображень, але таких можливостей нема в сучасних додатках, а є тільки наближені до такого рівня редактори, які не можуть виконати весь той функціонал який дає програмні продукти на ПК. Саме тому тема “Система виділення динамічних

об'єктів на серії зображень за допомогою нейронної мережі” допоможе зменшити зусилля для редагування і є актуальною на цей час.

## Розділ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАДАЧІ

### 1.1. Огляд існуючих рішень

Загальною ідеєю всіх рішень є одна й та сама а саме видалення не потрібних, що не вписуються в загальну картину зображення. Використовують для цього, вони, пару способів які відрізняються у здійсненні цих фотографій, Один спосіб змушує витратити меншу кількість сил на фотографування але при цьому збільшується час обробки фото, і якість обробленої фотографії залежить від фону. На другий спосіб витрачається більше зусиль на фотографію, але так само витрачаються зусилля на обробку фото але при цьому якість буде кращою. Далі приведено існуючі програми які демонструють ці способи на практиці.

#### 1.1.1. Remove

Не всім відома компанія Scalado розробила додаток для смартфонів. Додаток має назву “Remove” і вона видаляє з фотографії не потрібних для користувача людей.

Працює вона так: замість одного кадру робиться серійна зйомка (10 кадрів підряд) з проміжком 0,5-1 секунди, які зливаються в єдину фотографію. Статичні пікселі залишаються на місці, нестатичні - просто видаляються. Таким чином, всі перехожі і взагалі будь-які рухомі об'єкти просто зникають з фотографії.

Проблемою даного додатку є те, що з моменту оприлюднення можливостей “Remove” пройшло вже 6 років але додаток так і не вийшов у світ, також проблемою є витрата великого проміжку часу на створення фотографії.



Рис.1.1 Додаток Remove в дії

### 1.1.2. Adobe Photoshop Fix

Один з найпопулярніших та створений відомою компанією Adobe “Adobe Photoshop Fix”

Adobe Photoshop Fix - це програма для редагування фотографій для Android та iOS від Adobe Systems для ретушування. Він був представлений на Apple Keynote під час презентації iPad Pro та власної виставки Mac Max 2015

Photoshop Fix включає в себе безліч функцій, які також доступні у професійному Photoshop CC на робочому столі: корекція експозиції, контрастності, насиченості, глибини та освітлення за допомогою слайдера; Фотографії можна повертати, відображати та обрізати.

Істотна частина програми є інструментами ретушування, з допомогою яких зображення на Android мобільних пристроїв та iPhone або iPad є можливість відредагувати. Деякі функції редагування відрізняються від Photoshop та Lightroom. Загалом, існує 10 різних меню, кожне з яких має кілька підфункцій для редагування зображень. Класичні стандартні функції, такі як Штамп, корекція ефекту червоних очей і Spot Healing Brush, доповнюється інструментом, окремі частини можуть бути свідомо деформуються і адаптовані до його особам ретуші. Програмне забезпечення розпізнає обличчя в кадрі,

зазначений підборіддя, рот, очей і носа автоматично і дозволяє потім змінити, наприклад, ширину і довжину довести модель посміхатися або штучні очі, щоб збільшити або зменшити масштаб.

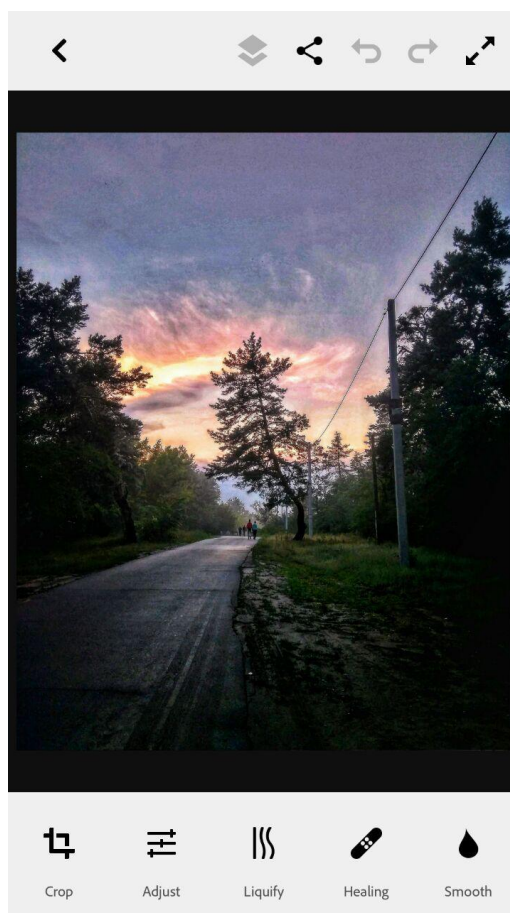


Рис 1.2 Приклад роботи Adobe Photoshop Fix

Робота додатка може бути зроблена пальцем по сенсорному екрану та за допомогою стилуса або з Apple Pencil. Фотографії, відредаговані програмою Adobe Photoshop Lightroom, можуть бути передані в Adobe Photoshop Fix для подальшої обробки.

### 1.1.3.Zoner Photo Studio

“Zoner Photo Studio” представляє собою програмне додаток, розроблене в Чеській founded компанії Zoner Software. Цей растровий редактор і менеджер файлів зображень є однією з найбільш широко використовуваних програм для редагування цифрових фотографій у своїй країні походження, а також широко використовується у всьому світі.

Ця лінія програмного забезпечення була спочатку опублікована під назвою Zoner Media Explorer. У 2004 році вона була перейменована в Zoner Photo Studio, оскільки фокус на продукт перейшов лише на суто цифрову фотографію. Щороку під час історії програм видається нова версія.

Це справді дуже хороші додатки в якому є багато можливостей, і в ній теж є можливість прибрати різні предмети з фотографії, але їх потрібно прибирати вручну і в ній немає автоматизації виділення людського силуету та видалення його.

## 1.2. Постановка Задачі

### 1.2.1. Об'єкт та предмет дослідження.

При фотографуванні різних краєвидів, будинків і т.д., на фотографії попадають багато рухомих об'єктів, щоб їх прибрати треба використовувати різні програми. У них є багато можливостей для редагування зображень, але таких можливостей нема в сучасних додатках, а є тільки наближені до такого рівня редактори, які не можуть виконати весь той функціонал який дає програмні продукти на ПК. Для цих завдань використовується «Штамп»/Clone Stamp Tool або «Заплата»/Patch Tool. Якщо зайвий об'єкт розташований на не однотонному фоні, то ці методи не є ідеальними, при обробці великої зони фотографії цими способами буде помітно не озброєним оком недоліки зробленої роботи. Також для цього потрібен ПК але це вже не практично, на це потрібно витратити багато часу, та навички, отже потрібен такий додаток який зможе впоратися з цією задачею.

Об'єкти, що містяться в зображенні, можуть бути розташовані та ідентифікуватися автоматично. Це називається виявлення об'єктів і є однією з основних проблем комп'ютерного бачення. Згорткові нейронні мережі в даний час є найсучаснішим рішенням для виявлення об'єктів. Методи виявлення згорткових об'єктів в останні роки покращилися. наскільки легко можна реалізувати систему детектування згорткових об'єктів наскільки добре система



виявляє об'єкти, навчання на загальних даних зображення, виконання певної задачі.

В данній роботі буде створена та навчена нейронна мережа для виявлення об'єктів на фото.

### 1.2.2. Переваги системи виділення динамічних об'єктів на серії зображень за допомогою нейронної мережі

Переваги системи виділення динамічних об'єктів на серії зображень за допомогою нейронної мережі котрі виділяють цю систему з поміж інших які можна перерахувати:

- Швидкість редагування зображення. Видаляючи рухомі об'єкти ми можемо витратити на це багато часу. Тому використовуючи потужність вашого смартфона є можливі використати нейронну мережу яка це все зробить за вас.
- Зручність використання котра також грає чималу роль у швидкості редагування фотографії, зручне розташування елементів управління ключ до цього успіху.
- Збирання статистики кількості зроблених фотографій, кількості видалених об'єктів.
- Використання нейронної мережі. З кожною фотографією зробленою через цей додаток нейронна мережа буде навчатися, і покращувати свою виконану роботу. Чим більше буде зроблено фотографій тим більша буде ефективність нейронної мережі.
- Можливість виправити невірно вирізані рухомі об'єкти з фотографії, чим більше виправлень тим менше їх зустрічатиметься в майбутньому.

### 1.2.3. Недоліками системи виділення динамічних об'єктів на серії зображень за допомогою нейронної мережі

Недоліками системи виділення динамічних об'єктів на серії зображень за допомогою нейронної мережі можна визначити такі:

- Щоб система вірно працювала потрібно деякий час простояти нерухомо або використовувати штатив.
- З цілей переходу користувачів на нові технології, додаток для мобільних телефонів підтримує п'яту та вище версію Android.
- За проблеми ранньої стадії розробки нейронна мережа буде часто помилятися

### Висновки по розділу

В цьому розділі було проаналізовано існуючі рішення для видалення об'єктів з фотографії такі як : “Remove” проблемою даного додатку є те, що з моменту оприлюднення можливостей “Remove” пройшло вже 6 років але додаток так і не вийшов у світ. “Adobe Photoshop Fix” робота додатка може бути зроблена пальцем по сенсорному екрану та за допомогою стилуса або з Apple Pencil. “Zoner Photo Studio” це хороший додаток яле він працює тільки на персональних комп'ютерах. Це справді дуже хороші додатки в якому є багато можливостей, і в ній теж є можливість прибрати різні предмети з фотографії, але їх потрібно прибрати вручну і в ній немає автоматизації виділення рухомих об'єктів та видалення їх. Якщо зайвий об'єкт розташований на не однотонному фоні, то ці методи не є ідеальними, при обробці великої зони фотографії цими способами буде помітно не озброєним оком недоліки зробленої роботи. Також для цього потрібен ПК але це вже не практично, на це потрібно витратити багато часу, та навички, отже потрібен такий додаток який зможе впоратися з цією задачею.

Описано мету системи виділення динамічних об'єктів на серії зображень за допомогою нейронної мережі. Багато методів, які вимагають реального часу виконання, можуть досягти цього лише за апаратної вартості тисяч євро. Розписано які можуть бути недоліки та переваги цієї системи.

## Розділ 2. Аналіз відомих нейронних мережі та вибір інструментів розробки

### 2.1. Штучна нейронна мережа

#### 2.1.1. Історія виникнення нейронної мережі

Людський мозок вивчається багато років. Після появи сучасних технологій з'явилася можливість спробувати застосувати процес мислення та навчання. Перший крок до створення нейронної мережі був здійснений в 1943 році, коли нейрофізіолог Уоррен Мак Куллох та молодий математик Вальтер Піттс написали статтю про те, як працюють нейрони. Вони моделювали просту нейронну мережу з електричними ланцюгами.

При появі комп'ютерів на початок 50-х років, стало можливим почати використати ці теоретичні знання на практиці. Натаніал Рочестер з дослідницьких лабораторій IBM провів перші експерименти, щоб імітувати нейронну мережу. Але за декількох невдач спроби пішли на лад та були успішними. Саме в той час традиційні обчислення почали цвісти, основна увага при обчисленні було надано нейронній мережі у фоновому режимі.

У 1959 році вчені з університета Стенфорд розробили алгоритми, які вони називають ADALINE і MADALINE. Ці алгоритми були названі з використанням декількох елементів ADaptive та LINEar. MADALINE була першою нейронною мережею яку застосували для вирішення проблем реального світу. Це адаптивний фільтр, який виключає відлуння на телефонних лініях. Ця нейронна мережа все ще знаходиться у комерційному використанні.

Вчений з Caltech опублікував статтю. Підхід полягав у тому, щоб не просто моделювати мозок, а створювати пристрої, алгоритми на основі роботи мозку для полегшення співіснування людей та технологій. Для ясності досліджуючи математичним аналізом він показав, як ці мережі можуть працювати і що вони можуть зробити. Однак одним з його талантів була харизма.

У 1982 р. відбулася чергова конференція в Кіото, Японія. Ця конференція була спільною з США та Японією з питань нейронних мереж. Згодом Японія оголосила про свої зусилля. Американські видання опублікували цю історію в люди, викликаючи занепокоєння, що США можуть залишитися на задньому плані з питань розвитку нейронної мережі. Незабаром фінансування оновилося.

1985 року в Американському інституті фізики відбулася перша нарада яка стала щорічною - Neural Networks for Computing. До 1987 року в Інституті інженерів з електротехніки та електроніки (IEEE) відбулася перша міжнародна конференція з нейронних мереж, яка налічувала понад 1800 учасників.

В останнє десятиліття про нейронну мережу можна почути всюди . Їхній потенціал виглядає дуже перспективним. В даний час розвиток нейронних мереж все більше укріплює впевненість в неймовірному потенціалі нейронних мереж.

### 2.1.2. Застосування нейронних мереж до різних сферах

Нейронні мережі широко використовуються для вирішення великої кількості реальних завдань, таких як прогнозування продажів, дослідження клієнтів, перевірка даних та управління ризиками.

- Маркетинг

Цільовий маркетинг передбачає сегментацію ринку, де ми поділяємо ринок на окремі групи клієнтів з різною поведінкою споживачів.

Нейронні мережі підготовлені до того, щоб вирішувати дійсні проблеми, фільтруючи клієнтів відповідно до основних характеристик, економічний стан, включаючи демографічні показники, місце розташування, моделі покупки та ставлення до продукту. Незабезпечені нейронні мережі можуть використовуватися для автоматичного групування та сегментації клієнтів на основі подібності їх характеристик, тоді як контрольовані нейронні мережі можуть навчатися, щоб вивчати межі між сегментами клієнтів на основі групи клієнтів.

- Роздрібні продажі

Нейронні мережі мають можливість одночасно розглядати кілька змінних, таких як ринковий попит на продукт, дохід клієнта, кількість населення та ціну продукту. Прогнозування продажів у супермаркетах може мати великі переваги.

Якщо існує зв'язок між двома продуктами впродовж деякого часу, скажім протягом 5-6 місяців після придбання автомобіля, клієнт повертається, щоб придбати новий повітряний фільтр для двигуна внутрішнього згорання, тоді торговці можуть використовувати цю інформацію для контакту з клієнтом, зменшуючи можливість покупки продукт в іншого продавця.

- Банківська справа та фінанси

Нейронні мережі успішно застосовуються до таких проблем, як геджування, цінних паперів, прогнозування, прогнозування курсу валют та ефективність акцій. Традиційно, програмне забезпечення керувало статистичними методами. Однак в наші дні нейронна мережа є основною технікою, що веде до прогнозування та прийняття рішень.

- Медицина

Це популярна дослідницька область в медицині, і вважається, що вони будуть отримувати широке застосування біомедичних систем в найближчі кілька років. На даний момент дослідження полягають в основному на моделюванні частин тіла людини та визнанні захворювань.

### 2.1.3. Нейронна мережа в роботі

Штучна нейронна мережа є одним з основних інструментів для машинного вивчення. Нейронна мережа складається з декількох вузлів, які імітують біологічні нейрони людського мозку. Нейрони зв'язуються по ланках, і вони взаємодіють один з одним. Вузли можуть приймати вхідні дані та виконувати прості операції з даними. Результат цих операцій переданий іншим нейронам. Вихід на кожному вузлі називається його активацією або значенням вузла. Нейронні мережі здатні до навчання, що відбувається шляхом зміни значення ваги [5]. Кожен нейрон помножує початкове значення на деяку вагу, суми результатів з іншими значеннями, що надходять в один і той же нейрон, налаштовує отримане число за зміщенням нейрона, а потім нормалізує результат за допомогою функції активації. На рис. 2.1 показано просту нейронну мережу.

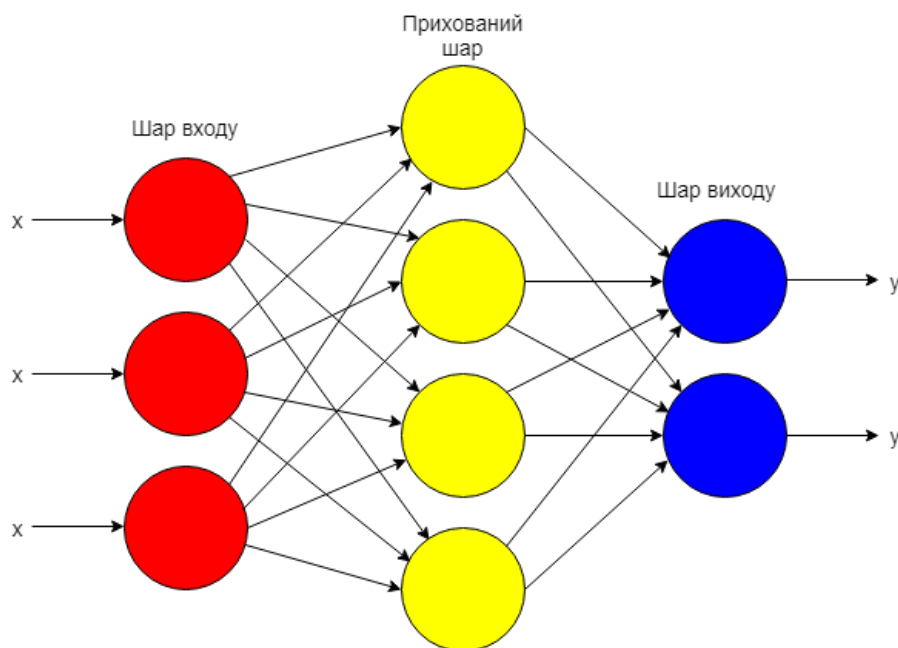


Рис. 2.1. Проста нейронна мережа

На сьогоднішній день існує багато способів розробки нейронної мережі для цього є такі види нейронних мереж: згорткова нейронна мережа, автокодувальник, рекурентна нейронна мережа.

Тому потрібно обрати найбільш продуктивну для нашої задачі нейронну мережу.

#### 2.1.4. Методи навчання нейронної мережі

Алгоритми навчання широко використовуються у програмах для комп'ютерного зору. Перш ніж розглядати задачі, пов'язані з зображенням, ми коротко розглянемо основи машинного навчання.

Алгоритми навчання стали корисним інструментом для моделювання проблем, які інакше можна точно сформулювати. Класичні комп'ютерні програми явно запрограмовані вручну для виконання завдання. З машинним навчанням частина людського внеску замінюється алгоритмом навчання. Оскільки наявність обчислювальних потужностей та даних зросла, навчання машинам стало більш практичним протягом багатьох років, до того, що вони майже повсюдно.

Типовим способом машинного навчання є навчання з вчителем. В алгоритмі навчання є кілька прикладів, які були анотовані або позначені людьми. Наприклад, у проблемі виявлення об'єктів ми використовуємо навчальні зображення, де люди позначені на фотографії на виділені іншим кольором. Навчившись з прикладів, алгоритм здатний дізнатися де знаходиться об'єкт на фото на фото які не входили до навчання з вчителем. Класифікація та регресія є найбільш важливими типами завдань. У класифікації алгоритм намагається передбачити правильний клас нового фрагмента даних на основі даних навчання. У регресії, замість дискретних класів, алгоритм намагається передбачити безперервний вихід.

У навчанні без вчителя алгоритм намагається вивчити корисні данні, але вказано, що має бути на виході. Класичним прикладом навчання без вчителя є кластеризація. Зовсім недавно, особливо з появою глибоких технологій навчання, навчання без вчителя попередня обробка стала популярним інструментом під контролем навчальних завдань для виявлення поданих даних.

Попередня обробка майже завжди потрібна. Переробка даних у нове, простіший змінний простір має назву витягом функцій. Один з десяти, це непрактично або неможливо безпосередньо використовувати повно розмірні

навчальні дані. Скоріше, детектори запрограмовані для того, щоб витягувати функції з даних, і ці функції використовуються як вхід до алгоритму машинного навчання.

У далекі часи детектори властивостей часто були створені ручним способом. Проблема з цим підходом полягає в тому, що ми не завжди знаємо заздалегідь, які функції є ефективними. Тенденція до машинного навчання полягала також в вивченні детекторів функцій, що дозволяє використовувати повні дані.

Оскільки дані навчання не можуть містити будь-які можливі екземпляри вхідних даних, алгоритм навчання повинен мати можливість узагальнювати для обробки невидимих точок даних. Занадто проста оцінка моделі не може зафіксувати важливі аспекти справжньої моделі. З іншого боку, надто складні методи можуть бути використані при моделюванні важких деталей та шумів, що також призводить до поганого результату. Як правило, зависання трапляється, коли складний метод використовується разом із занадто малими навчальними даними. Моделі, навчаються моделювати відомі приклади, але не розуміють, що їх з'єднує.

Продуктивність алгоритму можна оцінити за якістю та кількістю помилок. Функція втрат, така як середня квадрата помилка, використовується для призначення витрат на помилки. Метою навчального етапу є мінімізація цієї втрати.

## 2.2. Згорткова нейронна мережа

Об'єкти, що містяться в зображенні, можуть бути ідентифікуватися автоматично. Це називається розпізнавання об'єктів і є однією з основних проблем комп'ютерного бачення. Як буде продемонстровано, згорткові нейронні мережі в даний час є найсучаснішим рішенням для розпізнавання об'єктів.



Нейронна мережа - це поєднання нейронів. Нейрони, як правило, згруповані в шари. У повністю підключеній багатопотоковій мережі, показаному на рисунку 2.2, кожен вихідний сигнал шару нейронів надходить як вхідний до кожного нейрону наступного шару. Таким чином, деякі шари обробляють оригінальні вхідні дані, а деякі обробляють дані, отримані від інших нейронів. Кожен нейрон має ряд ваг, що дорівнює сумі ваг нейронів у попередньому шарі.

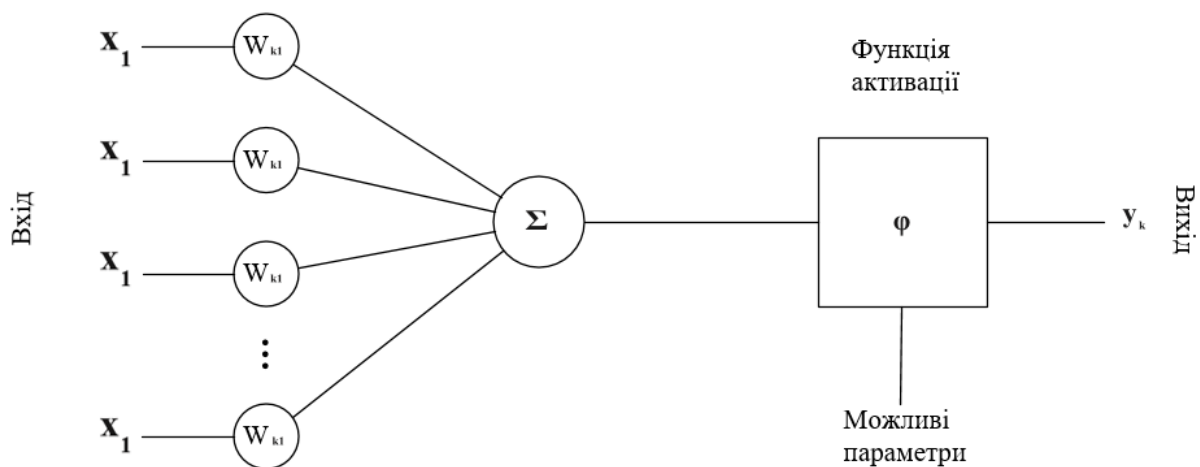


Рис. 2.2 Штучний нейрон

Багатошарова мережа зазвичай включає три типи шарів: вхідний шар, один або кілька прихованих шарів і вихідний шар. Вхідний шар зазвичай просто передає дані. Більшість обчислень відбувається в прихованих шарах. Вихідний рівень перетворює активацію прихованого шару на вивід. Багатошарова прохідна мережа з принаймні одним прихованим шаром може функціонувати як універсальний аппроксиматор, тобто може бути побудований для обчислення практично будь-якої функції.

Згорткові нейронні мережі дуже схожі на звичайні нейронні мережі: вони також збудований на основі нейронів, які володіють постійно змінюваних вагою і зміщення. Кожен нейрон отримує деякі вхідні дані, виконує скалярний добуток інформації і в окремих ситуаціях супроводжує це нелінійністю. Як і у випадку зі звичайними нейронних мережами, вся згорткова нейронна мережа висловлює одну диференційовану функцію внеску (ефективний внесок): з

одного боку це необроблені пікселі зображення, з іншого - висновок класу або групи ймовірних класів, які характеризують картинку. Тут також присутня функція втрати на останньому (повністю підключення) шарі. Простіше кажучи, згорткова нейронна мережа - це глибока модель навчання або багат шаровий перцептерон, подібний до штучних нейронних мереж, який найчастіше застосовується для аналізу візуальних образів.

Згорткова нейронна мережа отримала таке ім'я від "convolution". Основною метою є отримання функцій із вхідного зображення. Згортковість зберігає просторову зв'язок між пікселями шляхом вивчення функцій зображення за допомогою малих квадратів вхідних даних. Згорткова нейронна мережа була успішна в таких галузях, як:

- Розпізнавання обличчя

У процесі розпізнавання обличчя вони використовували каскад CNN для швидкого виявлення обличчя. Оцінюється вхідне зображення при низькій роздільній здатності, щоб швидко прибрати область без обличчя та обережно обробити складні області при більш високій роздільній здатності для точного виявлення потрібного об'єкта.



Рис. 2.3. Приклад роботи згорткової нейронної мережі

Також було введено калібрувальні сітки в каскад для прискорення виявлення та поліпшення якості роботи нейронної мережі.

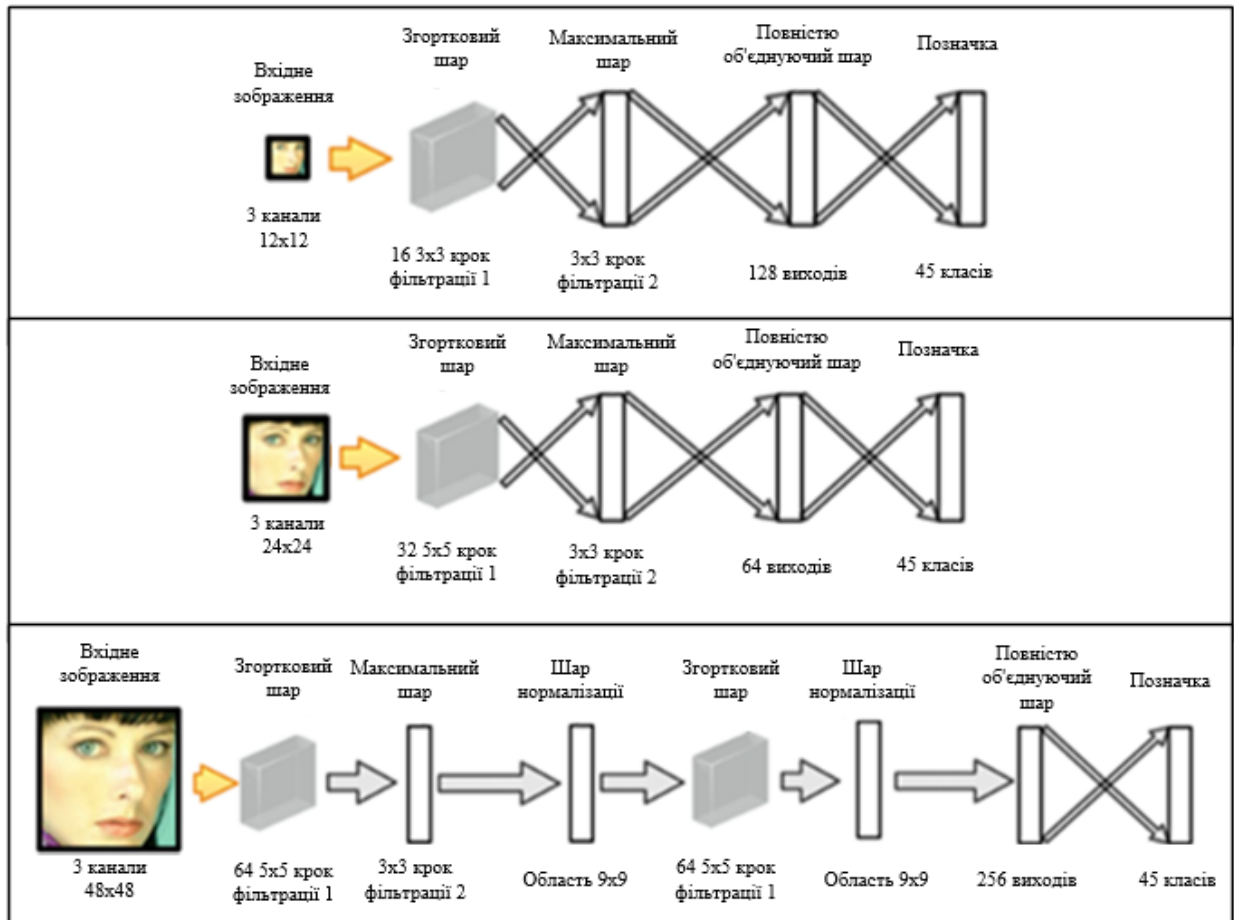


Рис. 2.4. Калібрувальні сітки

- Самохідні автомобілі

У проектах самокерованими автомобілями, оцінка навколишнього середовища є важливим аспектом в само керуючому автомобілі, оскільки це забезпечує безпеку пасажирів та інших транспортних засобів. Такі аспекти використання CNN були застосовані в таких проектах, як автономна машина NVIDIA.

Шляхи обробки CNN дозволяють їм бути надзвичайно універсальними, оскільки вони можуть обробляти дані за допомогою декількох параметрів. Згоркові нейронні мережі традиційно використовуються для аналізу зображень та розпізнавання об'єктів.

### 2.2.1. Основна структура

Основна ідея згорткової нейронної мережі надійшла від біологічної концепції, що називається рецептивним полем.

Вони діють як детектори, чутливі до певних типів подразників, наприклад, краї. Вони знаходяться на візуальному полі та перекривають один одного.

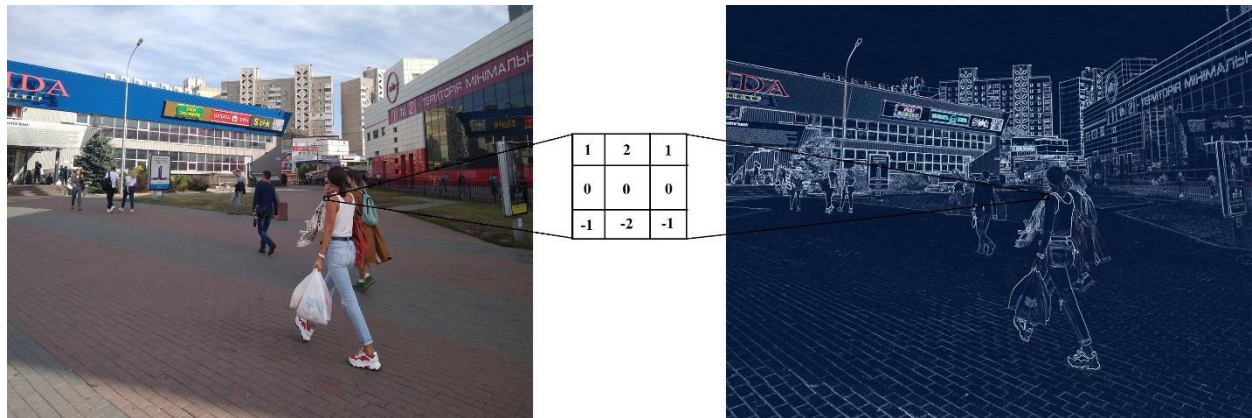


Рис. 2.5. Виявлення горизонтальних країв з зображення

Ця функція може бути апроксимована на комп'ютерах, використовуючи операцію усунення. Під час обробки зображень зображення можна згортати, використовуючи згортку, щоб створювати різноманітні видимі ефекти. На рисунку 2.5 показано, як виділений вручну виявляє горизонтальні краї зображення, що функціонує аналогічно рецептивному стану.

Дискретна операція згортки між зображенням  $f$  та матрицею фільтрів  $g$  визначається як:

$$h[x, y] = f[x, y] * g[x, y] = \sum_n \sum_m f[n, m] g[x - n, y - m].$$

По суті точка обробки фільтра  $g$  та суб-образ  $f$  (з такими ж розмірами, як  $g$ ), орієнтовані на координати  $x, y$ ; виробляє піксельне значення  $h$  за координатами  $x, y$ . Розмір рецептивного поля регулюється розміром матриці фільтра. Вирівнювання фільтра послідовно з кожним суб-зображенням  $f$  дає вихідну піксельну матрицю  $h$ . У випадку нейронних мереж вихідна матриця також називається картою властивостей (або карта активації після обчислення функції активації). Краї слід розглядати як особливий випадок. Якщо

зображення  $f$  не заповнено, розмір вихідного зображення зменшується з кожною згорткою.

Сукупність згорткових фільтрів може бути об'єднана, щоб сформувати згортковий шар нейронної мережі. Матричні значення фільтрів розглядаються як параметри нейронів і навчаються за допомогою машинного навчання. Операція згортки замінює операцію множення звичайного нервового мережевого рівня. Вихідний шар зазвичай описується як об'єм. Висота та ширина обсягу залежать від розмірів карти активації. Глибина обсягу залежить від кількості фільтрів.

Оскільки ті ж самі фільтри використовуються для всіх частин зображення, кількість вільних параметрів різко зменшується в порівнянні з повністю з'єднаним нейронним шаром. Нейрони згорткового шару в основному поділяють однакові параметри і пов'язані лише з локальним регіоном входу. Розподіл параметрів в результаті згортки забезпечує інваріантність перекладу. Альтернативним способом опису згорткового шару є повністю з'єднаний шар з нескінченно сильним попередньо поставленим на ваги. Це попереднє змушує нейрони розподіляти ваги в різних просторових місцях і мати нульову вагу поза рецептивним полем.

Послідовні згорткові шари (часто поєднуються з іншими типами шарів, так звані об'єднання) утворюють згорткову нейронну мережу (CNN). Приклад згорткової мережі показано на рис. 2.6. У теорії, шари, ближчі до входу, повинні навчитися розпізнавати низькорівневі функції зображення, наприклад, краї та кути, а шари ближче до виводу повинні навчитися об'єднувати ці функції, щоб розпізнати більш значущі об'єкти. У цій темі ми зацікавлені у вивченні того, чи можуть згорткові мережі навчитися розпізнавати повні об'єкти.

Глибокі шари неронної мережі потребують менше вхідної інформації про точні розташування функцій, і вимагають більш легких матриць для розпізнавання декількох моделей високого рівня. Знижуючи висоту та ширину об'єму даних, ми можемо збільшити глибину обсягу даних.

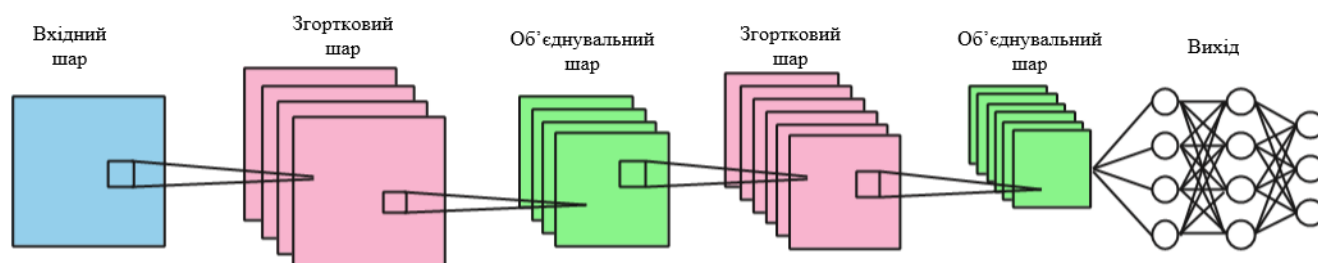


Рис. 2.6. Приклад згорткової мережі

Існує два способи зменшення обсягу даних. Один із способів полягає в тому, щоб включити згрупований шар після згорткового. Поєднання має додатковий ефект, що робить отриману мережу більш інваріантною, змушуючи детектори бути менш точними. Однак об'єднання даних може стерти інформацію про просторові зв'язки між підрозділами шаблонів. Типовий метод згрупування - максимальне об'єднання. Максимальне об'єднання просто виводить максимальне значення в прямокутній карти активації.

Іншим способом зменшення розміру об'єму даних є коригування масштабу вимірювального сигналу операції згортки. Параметр “крокування” визначає, чи обчислюється висновок згортки для окремої межі, орієнтованої на кожен піксель вхідного зображення (кроки 1) або для кожного  $n$ -го пікселя (кроки  $n$ ). Дослідження показали, що з'єднувальні шари часто можна відкидати без втрати точності, використовуючи згорткові шари з великим значенням кроку. Операція “крокування” еквівалентна використанню сітки для об'єднання.

### 2.2.2. Додаткові шари рекурентної нейронної мережі

Згортковий шар зазвичай включає в себе нелінійну функцію активації, таку як функція прямолінійної активації. Активация іноді описується як окремий шар між згортковим шаром і об'єднуючим шаром.

Деякі системи, такі як [2]. Також реалізують шар, що називається Local Response Normalization(LRN), який використовується як метод регуляризації. LRN імітує функцію біологічних нейронів, що називається “lateral inhibition”, що змушує збуджувані нейрони зменшувати активність сусідніх нейронів. Проте інші методи регуляризації в даний час більш популярні.

Приховані шари CNN, як правило, повністю пов'язують шари. Повністю з'єднаний шар може захоплювати деякі цікаві з'єднання, поділ параметрів зі зсувних шарів не може бути. Однак повноцінний шар вимагає достатньо невеликих обсягів даних, щоб бути практичним. Параметри збірки та кроків можуть бути використані для зменшення об'єму даних, який досягає повністю підключених шарів. Згорткова мережа, яка не містить жодних повноз'єднаних шарів, називається “Fully Convolutional Network” (FCN).

Якщо мережа використовується для класифікації, вона звичайно включає в себе вихідний рівень. Активації верхніх шарів також можуть бути використані безпосередньо для генерування представлення об'єкта зображення. Це означає, що згорткова мережа використовується як великий детектор властивостей.

### 2.2.3. Регуляризація та збільшення даних

Регуляризація відноситься до методів, які використовуються для зменшення витрат шляхом введення додаткових обмежень або інформації в систему машинного навчання. Класичним способом регуляризації в нейронних мережах є додавання строку покарання до функції об'єктивної втрати. Іншим прикладом регуляризації є функція обміну параметрами згорткових мереж.

Існує кілька методів регуляризації, які є специфічними для глибоких нейральних мереж. Популярна методика, що називається відсів, намагається зменшити коадаптацію нейронів. Це досягається випадковим випаданням нейронів під час навчання, а це означає, що для кожної навчальної вибірки або

мініатюру використовується трохи різноманітна нейронна мережа. Це призводить до того, що система не занадто сильно залежить від жодного окремого нейрона або з'єднання і забезпечує ефективний, але ефективний спосіб здійснення регуляризації. У згорткових мережах, відсів, як правило, використовується у повних з'єднаних шарах.

Також можна зменшити витрати, збільшуючи кількість навчальних даних. Коли неможливо знайти більш актуальні зразки, збільшення даних використовується для отримання більшої кількості зразків з існуючих даних. Для класифікації, використовуючи конволюційні мережі, це може бути досягнуто шляхом поєднання перетворень вхідних зображень, які не змінюють класів сприйманих об'єктів, але надають додаткові виклики системі. Зображення можуть бути, наприклад, обернені або згруповані різними вагами. Крім того, шум може бути доданий до вхідних даних.

#### 2.2.4. Виявлення згорткового об'єкта

- R-CNN

Обчислення R-CNN на ранніх стадіях має кілька етапів, показаних на рис. 2.7. По-перше, створюються області. Трансформуються в незалежні від категорій обмежувальні контури, які мають високу ймовірність містити потрібні об'єкти. У роботі для створення цих методів використовується окремий метод, що називається “Selective Search”, але замість нього можуть використовуватися інші методи генерації області. Вибірковий пошук разом із методами генерації пропозицій інших областей.

Далі для здобуття функцій із пропозицій кожної області використовується згорткова мережа. Суб-зображення, що міститься в контурах, деформується, щоб відповідати розміру вхідного файлу CNN, а потім надсилає його в мережу. Після того, як мережа витягнула функції з введення, функції вводяться для підтримки векторних машин (SVM), які забезпечують остаточну класифікацію.





Рис. 2.7 Етапи раннього обчислення R-CNN

R-CNN є важливим методом, оскільки він забезпечив перший практичний розв'язок для виявлення об'єктів за допомогою CNN. Будучи першим, він має багато недоліків, які були вдосконалені згодом.

У документі 2015 для Fast R-CNN [7] є три основні проблеми R-CNN.

По-перше, навчання складається з декількох етапів, як описано вище. По-друге, навчання є дорогим. Для навчання щодо пропозиції SVM та областей функції витягуються з пропозицій кожної області та зберігаються на диску. Для цього потрібні дні обчислень і сотні гігабайт пам'яті.

По-третє, і, мабуть, найважливіше, виявлення об'єктів є повільним, що вимагає майже хвилини для кожного зображення, навіть на графічному процесорі. Це відбувається тому, що обчислення вперед CNN виконується окремо для кожної пропозиції об'єкта, навіть якщо пропозиції походять з одного зображення або перекривають один одного.

- Швидкий R-CNN

Швидкий R-CNN опублікований у 2015р. Компанією Girshick, надає більш практичний спосіб розпізнавання об'єктів. Головна ідея полягає в тому, щоб виконати прямий прохід CNN для всього зображення, а не виконувати його окремо для кожного зони.

Загальна структура швидкого R-CNN зображено на рис. 2.8. Метод отримує в якості вхідних даних зображення плюс області, що представляють інтерес, обчислених з зображення. Як і в R-CNN, Область інтересу створюються за

допомогою зовнішнього методу. Образ обробляється за допомогою CNN, який включає в себе декілька згорткових і максимально об'єднаних шарів.

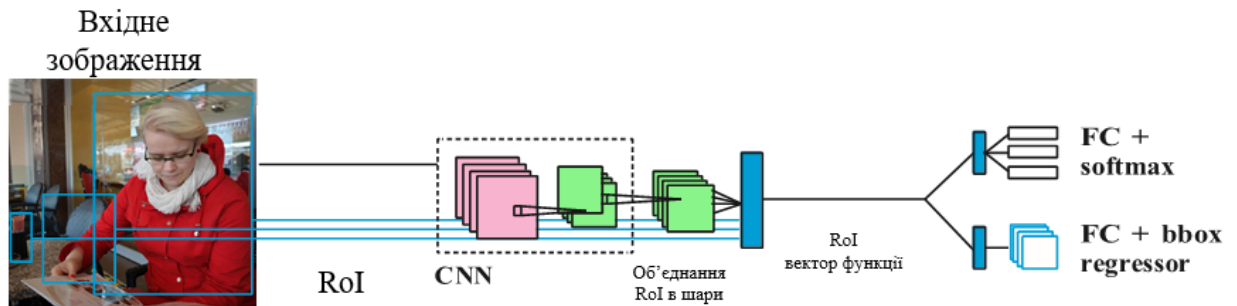


Рис. 2.8. Етапи швидкого розрахунку R-CNN

Карта згорткових властивостей, яка генерується після того, як ці шари вводяться в шар об'єднання. Це витягує вектор-функцію фіксованої довжини для кожної області інтересу з карти об'єкта. Вектори властивостей потім вводяться до повністю підключених шарів, які підключені до двох вихідних шарів: рівень нормованої експоненційної функції, який створює оцінки ймовірності для класів об'єктів та реальний шар, який виводить кореляції країв, обчислену за допомогою регресії (тобто ознаки *regions* до початкового вікна кандидата).

#### 2.2.5. Продуктивність R-CNN

На думку авторів, швидкий R-CNN, забезпечує значно коротший час класифікації в порівнянні зі звичайним R-CNN, приймаючи менше секунди на найсучасніших GPU. Це пов'язано, головним чином, з використанням однієї карти функцій для кожної області інтересу.

Оскільки час виявлення зменшується, загальний час обчислення починає суттєво залежати від ефективності методу генерації пропозиції області. Таким чином, області інтересу, можуть сформувати обчислювальний вузький рівень. Крім того, коли існує безліч областей, час, витрачений на оцінку повністю підключених шарів, може домінувати час оцінки шарів. Час класифікації може бути прискорено приблизно на 30%, якщо цілком з'єднані шари стискаються,

використовуючи декомпозицію з усіченими однозначними значеннями. Однак це призводить до незначного зменшення точності.

Згідно з оригінальним виданням [7]. Швидкий R-CNN більш підготовлений, ніж R-CNN, з дев'ятикратним скороченням часу навчання. Вся мережа може пройти навчання за допомогою алгоритму зворотного поширення та дезактивації стохастичного градієнта. Як правило, попередньо підготовлена мережа використовується як вихідна точка, а потім не налаштовується. Навчання проводиться в міні-партіях з  $N$  образів.  $R = N$ . Відібрані області відбираються з кожного міні-партійного зображення. Зразки області інтересу призначаються для класу, якщо їх перетинання над об'єднанням з полем на істині становить більше 0,5. Інші області інтересу належать до фонового класу.

Як і в класі катіону, область інтересу з одного і того ж зображення поділяють обчислення та використання пам'яті. Для збільшення даних оригінальне зображення імпортується горизонтально з ймовірністю 0,5. Класифікатори нормованої експоненційної функції та регресери обмежувальної коробки об'єднуються разом, використовуючи багатозадачну функцію втрат, яка консервує як справжній клас зразка області інтересу, так і о зразок вибіркового обмежувального вікна з істинної обмежувальної коробки.

#### 2.2.6. Створення та використання пропозиції

Щоб використовувати R-CNN та Fast R-CNN, нам потрібен спосіб генерації областей класів-агностиків. Далі ми збираємося обговорити загальні принципи пошуку області інтересу і ближче подивимося на два популярних методи: Selective Search та Edge Boxes.

Метою генерації пропозиції регіону в процесі виявлення об'єктів є максимізація відкликання, тобто створення достатньої кількості областей для відновлення всіх реальних об'єктів

Щільний набір рішень генерується шляхом грубої сили вичерпний набір обмежувального поля, що включає кожне потенційне місце розташування об'єкта. Цього можна досягти, скориставшись виявлення по всьому зображенню. Проте потрібна велика потужність комп'ютера і вимагає швидкого детектора об'єктів. Крім того, повинні бути розглянуті різні форми і розміри. Таким чином, більшість методів зсуву обмежують кількість об'єктів-кандидатів, використовуючи великий розмір кроку та обмежену кількість хедових аспектних співвідношень.

Більшість пропозицій в наборі не містять потрібних об'єктів. Вони повинні бути відкинуті після фази виявлення об'єкта. Результати відхилення можуть бути відкинуті, якщо вони відстають від порогового значення, що не визначається, або якщо значення їх констанції є нижчим за локальний максимум (не максимальне придушення).

Одним з найпопулярніших безумовних методів є Selective Search, який використовує ітеративне злиття суперпікселів. Існують також інші методи, які використовують той самий підхід. Інший підхід полягає в оцінці об'єктивності розсувного поля. Популярним прикладом цього є Edge Boxes, який обчислює оцінку об'єктивності шляхом обчислення кількості ребер всередині обмежувальної коробки та вирахуванням кількості ребер, які перекривають межу коробки. Існує також третя група методів на основі сегментації насіння.

Наглядова методика розглядає генерацію пропозиції області як класифікацію або проблему регресії. Це означає, що використовується алгоритм машинного навчання, наприклад вектор машини підтримки. Також можна використовувати згорткову мережу, щоб генерувати цікаві облатсі. Прикладом використання CNN для розрахунку обмежувальних ящиків є Multi-Box.

Деякі методи розпізнавання додаткових об'єктів, наприклад, швидкий R-CNN, використовують частини тієї самої згорткової мережі як для створення

пропозицій області, так і для виявлення. Ми називаємо ці види методів інтегрованими методами.

### 2.2.7. Selective Search

Selective Search (Вибірковий пошук) використовує ієрархічне розділення зображення для створення розрідженого набору місць розташування об'єктів. Основна філософія полягає не в тому, щоб використовувати єдину стратегію, а об'єднати найкращі функції сегментації знизу-вгору та вичерпного пошуку. Автори мали три основні конструктивні міркування: пошук повинен охоплювати всі масштаби, бути різноманітними, тобто не використовувати єдину стратегію для групування областей і бути швидким для обчислення.

Алгоритм починається шляхом створення набору малих початкових областей за допомогою методу, що називається графічною сегментацією зображення. Метод створює набір регіонів, які називаються суперпікселями. Суперпікселі внутрішньо майже одноманітні. Поєднаними, вони охоплюють весь образ, але індивідуально вони не повинні охоплювати різноманітні об'єкти.

Вибірковий пошук потім продовжується, ітеративно згрупувавши області за допомогою алгоритму, починаючи з двох найбільш подібних областей. Для розрахунку подібності використовуються численні заходи. Ці заходи враховують схожість кольорів (шляхом обчислення кольорової гистограми), схожість тексту (шляхом обчислення межі подібної до SIFT), розміру області і прогалини повинні бути уникнути Фаза групування завершується, коли кожна область об'єднана.

Гіпотетичні позиції об'єктів, отримані таким чином, потім упорядковуються за вірогідністю розташування, що містить об'єкт. На практиці розташування розташовуються за порядком, в якому вони були згруповані разом різними заходами. Додано певний елемент випадковості, щоб запобігти надмірній популярності великих об'єктів. Видалені дублікати нижчого рангу.

Метод генерування регіону та заходи подібності були виділені для швидкого обчислення, що робить метод швидким у цілому. Окрім застосування різноманітних заходів щодо подібності, пошук може бути додатково різноманітним, використовуючи додаткові кольорові простори (для забезпечення інваріантності освітлення) та використання додаткових стартових регіонів.

#### 2.2.8. Edge Boxes

Як випливає з назви, Edge Boxes базується на виявленні об'єктів з краєвих карт. Основним внеском авторів методу є спостереження, що число контурів краю, повністю замкнених обмежувальним коробкою, корелює з ймовірністю того, що коробка містить об'єкт.

По-перше, карта краю розраховується за методом тих же авторів, що називають Структурований крайовий детектор. Тоді товсті лінії краю розріджуються з використанням максимального придушення. Замість того, щоб безпосередньо працювати на крайніх пікселях, пікселі групуються за допомогою алгоритму.

Регіональні пропозиції знайдені шляхом сканування зображення за допомогою традиційного методу розсувного вікна та розрахунку оцінок об'єктивності в кожному положенні, співвідношенні і масштабі. Оцінка розраховується шляхом підсумовування межі краю груп, які лежать повністю в коробці, і підкреслюють міцність груп краю, які є частиною контуру, який перетинає границя коробки.

#### 2.3. Автокодувальник

Цей підхід базується на спостереженні, попереднє навчання кожного шару за допомогою Випадкового алгоритму навчання може забезпечити кращу початкову вагу. Приклади таких алгоритмів - мережі глибоких переконань. Є

кілька недавніх спроб досліджень цієї області, наприклад, використання варіаційних методів для імовірнісних автокодувальних.

Вони рідко використовуються в практичних цілях. Нещодавно пакетна нормалізація дозволила надати ще більш глибоко навчені мережі, ми могли б довільно навчати глибокі мережі з нуля, використовуючи залишкове навчання. З відповідними обмеженнями на розмірність і обмеженість, автокодувальники можуть вивчати проєкційні дані, які є більш цікавими, ніж PCA або інші основні методи.

Давайте подивимося на два цікаві практичні застосування автокодувальників:

- У даних, що виводять шлюзу, автокодувальник, що вимикає, використовує згорткові шари, використовується для ефективного відтворення медичних зображень.

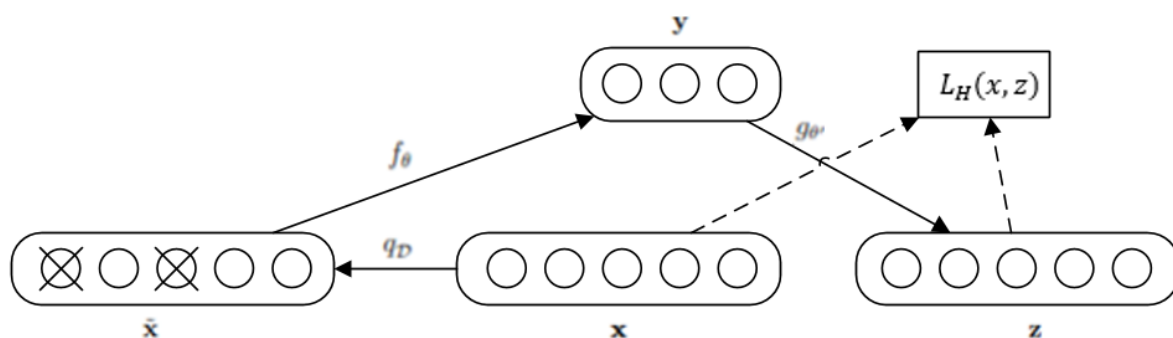


Рис. 2.9. Базова архітектура автоасоціаторов видалення шуму

Стохастичний процес випадковим чином встановлює деякі вхідні дані до нуля, змушуючи декодер, що відміряє, прогнозувати відсутні (пошкоджені) значення для випадково вибраних підмножин відсутніх візерунків.

- Зменшення розміру для спроб візуалізації даних зменшення розміру за допомогою таких методів, як аналіз принципів компонентів (PCA) та t-розподілений вбудований стохастичний сусід (t-SNE). Вони були використані в поєднанні з підготовкою нейронних мереж, щоб підвищити точність прогнозування моделі. Також, точність прогнозування нейронної мережі MLP

сильно залежить від архітектури нейронної мережі, попередньої обробки даних та типу проблеми, для якої була розроблена мережа.

## 2.4. Рекурентні нейронні мережі

Рекурентні нейронні мережі - RNN можуть бути застосовані для генерації (передбачення) послідовності шляхом обробки реальних послідовностей даних на один крок за один раз і прогнозування або передбачення наступних дій в багатьох сферах.

Припускаючи, що прогнози є ймовірнісними і можуть бути покращені з більшою кількістю вхідних даних, нові навчальні послідовності можуть бути сформовані з навченої мережі шляхом ітераційного відбору з вихідного розподілу мережі, а потім подання як вхід на наступний крок. Іншими словами, створюючи мережу вона розглядається так, як реальний предмет, подібно до того, як людина прогнозує оброблюючи деяку кількість даних.

Є можливість навчити рекурентну нейронну мережу генерувати рукописний текст. Для здійснення цього завдання ввідні данні об'єднуються з текстовим рядком і подає як додатковий данні до мережі прогнозування. Параметри виводяться мережею одночасно з прогнозуванням, так що він в реальному часі визначає вирівнювання між текстом та розташування пера. Простіше кажучи, він навчається вирішувати, який текст буде написаним далі.

Нейронна мережа може бути підготовлена для передбачення результатів в будь якій сфері, з урахуванням вхідних даних. Якщо у нас є мережа, яка добре підходить для моделювання послідовності значень, її можна використовувати для прогнозування майбутніх результатів. Очевидним прикладом є прогноз фондового ринку.

## 2.5. Обґрунтування вибору програмних засобів



### 2.5.1.Java

Java є мовою програмування розроблена Джеймсом Гослінг та іншими членами команди: Майк Шерідан і Патрік Нотон, також називають Green Team. В 1995 році Sun Microsystems для цифрових пристроїв: телевізійні приставки, телевізори і т.д. Тепер детальніше, що таке Java.

Це об'єктно-орієнтована мова, подібна до C ++, але спрощене при цьому маючи той самий функціонал функціями. Java має вільний доступ і може працювати на всіх платформах.

- Паралельність, де ви можете виконувати кілька операцій за один раз, а не виконувати їх послідовно.
- Об'єктно-орієнтована мова програмування.
- Незалежна мова програмування, що слідує за логікою " Напиши один раз, Запустити скрізь ", тобто скомпільований код може працювати на всіх платформах, що підтримує java машину.

Java використовується в різних сферах. Такі як:

- Банки: для вирішення питань управління транзакціями.
- Торгівля: додатки для сплати послуг, такі як в магазині, ресторані.
- Інформаційні технології: Java призначений для вирішення залежностей.
- Android: додатки або написані на Java або використовуються Java API.
- Фінансові послуги : сервера написані на Java.
- Фондовий ринок: написати алгоритми які рекомендують в які компанії потрібно вкладати.
- Великі дані: структура Hadoop MapReduce написана з використанням Java.
- Науково-дослідне співтовариство: для вирішення величезної кількості даних.

### 2.5.2.JSON

JSON є простим, текстовим редактором. Використовуючи простий синтаксис, ви можете легко зберігати одне число або безліч рядків, масивів і об'єктів, не використовуючи нічого, крім рядка звичайного тексту. Ви можете також створювати масиви, що дозволяє будувати висококласні структури.

Створивши рядок JSON, є можливість без жодних зусиль відправити в інший даток або комп'ютер, тому що це просто звичайні текстові данні.

JSON має багато переваг:

- Його можна легко поділити на структури даних, також мови програмування підпримуються у велика кількість.
- Майже всі мови програмування містять функцію або бібліотеки, які можуть читати та записувати у JSON.
- JSON розшифровується як JavaScript Object Notation. Що впливає з назви, це засноване на тому як позначаються.
- JSON розповсюджений у використанні веб-додатках для зв'язку між сервером та веб додатком.

### 2.5.3.Kotlin

Це статично типізована мова програмування, що працює на віртуальній машині Java, також може бути скомпільована в вихідний код JavaScript. Вона розроблена в основному компанією JetBrains. Хоча синтаксис несумісний з Java, реалізація JVM стандартної бібліотеки Kotlin призначена для взаємодії з кодом Java і залежить від коду Java від існуючої бібліотеки класів Java, наприклад, у рамках колекції. Kotlin використовує висновок агресивного типу, щоб визначити тип значень та виразів, для яких тип не був зазначений. Що стосується Android Studio 3.0 (випущений в жовтні 2017 року), Kotlin повністю підтримує Google для використання в операційній системі Android і безпосередньо входить до пакету інсталяції IDE в якості альтернативи

стандартному компіляторі Java. Компілятор Android Kotlin дозволяє користувачеві вибирати між Java 6, Java 7 або Java 8-сумісний байт-код.

#### 2.5.4.Phyton

Python - це інтерпретована мова програмування високого рівня, яка на сьогодні дуже швидко підіймається в рейтингах. Далі буде вказано плюси і мінуси Python і побачимо, де його використання буде вигідним.

Недоліки Python:

Порівняно з іншими мовами програмування Python останнім часом часто використовується розробниками. Основними перевагами мови Python є те, що вони легко читаються та легко вивчаються. Простіше писати програму в Python, ніж у C або C ++. За допомогою цієї мови ви отримуєте можливість чітко продумати під час кодування, що також полегшує підтримку коду. Це знижує витрати на підтримку програми та розглядається як один із переваг програми Python.

Python має ряд унікальних характеристик, які цінні для програмістів, оскільки вони полегшують кодування. Ще однією перевагою програмного забезпечення Python є те, що не може виникнути помилка сегментації.

Важливою перевагою мови Python є те, що вона широко застосовується і широко використовується вченими, інженерами та математиками. Саме з цієї причини Python настільки корисний для прототипів і всіх видів експериментів. Він використовується у багатьох новаторських полях. Він також використовується при створенні анімації для кінофільмів та машинного навчання.

Мова включає в себе велику бібліотеку з управлінням пам'яттю, що є ще однією перевагою мови Python.

#### 2.5.5.Camera API

У системі Android передбачено підтримку різних функцій камер та функцій камери, доступних на пристроях, що дозволяє робити знімки та відео у ваших програмах.

У більшості пристроїв Android є принаймні одна камера. Деякі пристрої мають передню та задню камеру. Використання камери на пристрої Android можна здійснювати за допомогою інтеграції існуючої програми камери. У цьому випадку ви мали б запустити існуючу програму для камери за допомогою наміру та використовувати дані повернення програми для доступу до результату.

Android SDK підтримує зв'язок із вбудованою камерою. Використання фотоапарата для зйомки порівняно легко. Дещо важче настроїти попередній перегляд камери для правильної роботи.

У нашій основній діяльності ми створюємо об'єкт попереднього перегляду. Цей об'єкт створить об'єкт Camera і поверне його в активність

Далі ми реєструємо пару методів зворотного зв'язку з Камерою, яка виконується, коли користувач бере фотографію. Крім того, ви можете безпосередньо інтегрувати камеру в свою програму за допомогою Camera API.

## 2.6. Розробка Android додатку

### 2.6.1. Інструменти розробки

- UML 2.0 (Unified Modeling Language – «уніфікована мова моделювання») Метою UML є надання стандартних позначень, які можуть бути використані всіма об'єктно-орієнтованими методами, а також для вибору та інтеграції найкращих елементів попередніх позначень. UML розроблений для широкого кола додатків. Отже, він забезпечує конструкції для широкого кола систем та діяльності (наприклад, розподілених систем, аналізу, проектування та розгортання системи).
- Java Development Kit 1.8, скорочено JDK — Комплект розробки Java (JDK) - це один з трьох основних пакетів технологій, що

використовуються в програмуванні Java, разом із JVM (Java Virtual Machine) і JRE (Java Runtime Environment). Важливо відрізнити ці три технології, а також розуміти, як вони пов'язані:

- JVM - компонент платформи Java, який виконує програми.
  - JRE - частина на диску Java, яка створює JVM.
  - JDK дозволяє розробникам створювати Java-програми, які можуть бути виконані JVM і JRE.
- Android SDK – Набір програмного забезпечення для розробників, який дозволяє розробникам створювати додатки для платформи Android. SDK для Android містить вибіркові проекти з вихідним кодом, інструментами розробки, емулятором та необхідними бібліотеками для створення додатків Android. Програми записуються за допомогою мови програмування Java і запускаються на Dalvik, користувальницькій віртуальній машині, призначеній для вбудованого користування, що працює над ядром Linux.
  - Kotlin Це статично типізована мова програмування, що працює на віртуальній машині Java, а також може бути скомпільована в вихідний код JavaScript . Вона розроблена в основному компанією JetBrains. Хоча синтаксис несумісний з Java, реалізація JVM стандартної бібліотеки Kotlin призначена для взаємодії з кодом Java і залежить від коду Java від існуючої бібліотеки класів Java , наприклад, у рамках колекції. Kotlin використовує висновок агресивного типу, щоб визначити тип значень та виразів, для яких тип не був зазначений. Що стосується Android Studio 3.0 (випущений в жовтні 2017 року), Kotlin повністю підтримує Google для використання в операційній системі Android і безпосередньо входить до пакету інсталяції IDE в якості альтернативи стандартному компіляторі Java. Компілятор Android Kotlin дозволяє користувачеві вибирати між націлюванням на Java 6, Java 7 або Java 8-сумісний байт-код.

### 2.6.2. Програмні пакети

- Adobe Photoshop CS6 13.0 – Зображення редагування програмне забезпечення розроблено і виготовлено Adobe Systems Inc . Photoshop вважається одним з лідерів у програмному забезпеченні для редагування фотографій. Програмне забезпечення дозволяє користувачам маніпулювати, обрізати, змінювати розмір та коректувати кольори на цифрових фотографіях. Програмне забезпечення особливо популярне серед професійних фотографів та графічних дизайнерів.
- Microsoft Visio 2016 PRO – векторний графічний редактор, редактор діаграм і блок-схем для Windows;
- Sublime Text 3 — це текстовий редактор, написаний на C ++ та Python, доступний у Windows, Mac та Linux. Текстовий редактор - це розробники програм, які пишуть свій код у.
- MySQL Workbench 6.3 CE — це система управління реляційною базою даних, яка дозволяє зберігати інформацію в кількох таблицях, що мають взаємозв'язок. Наприклад, таблиця «Клієнт» може містити дані про клієнтів разом із ключовим полем, який з'єднує таблицю із таблицею продажу, яка містить критичні дані про продаж. Ваша програма може зберігати інформацію в цих таблицях та отримувати її за потреби.
- Android Studio 2.3.2 — офіційне інтегроване середовище розробки ( IDE ) для розробки додатків для Android. Вона заснована на IntelliJ IDEA , інтегрованому середовищі розробки Java для програмного забезпечення, і включає в себе редагування коду та інструменти розробника.

### Висновок по розділу

Нейронні мережі можуть дати нам певне уявлення про те як мозок працює, як інтегрує інформацію. Але справжнє питання полягає в тому, чому і як вся ця обробка в людині супроводжується досвідченим внутрішнім життям і

чи може штучна нейронна мережа досягти такого рівня як біологічна нейронна мережа яка є в людині з народження?

Це змушує нас задуматися, чи можуть нейронні мережі стати пензлем для художників - новим способом створення, візуальні концепції, або, можливо, навіть пролити світло на коріння творчого процесу в цілому.

Загалом, нейронні мережі зробили комп'ютерні системи більш корисними, роблячи їх більш схожими до біологічної нейронної мережі людини. Таким чином, наступного разу, коли ви думаєте, що вам може сподобатися, то ваш мозок буде настільки ж надійним, як комп'ютер, подумайте ще раз будьте вдячними, що у вас є така чудова нейронна мережа, яка вже встановлена у вашій голові!

Протягом останнього десятиліття в області машинного навчання домінують так звані глибокі нейронні мережі, які користуються перевагами покращення в обчисленні потужності та доступності даних. Підтип нейронної мережі має назву згорткова нейронна мережа, добре підходить для завдань пов'язаних з зображенням. Мережа навчається шукати різні функції, такі як краї, кути та кольорові відмінності, та об'єднати їх у більш складні фігури. Для виявлення об'єкта система має оцінювати розташування імовірних об'єктів та їх класифікувати.

## Розділ 3. РОЗРОБКА АЛГОРИТМІЧНОГО ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

### 3.1. Експеримент згорткової нейронної мережі

Починаємо обговорення експериментальної частини дисертації. По-перше, ми обговоримо критерії відбору для методів і наборів даних. Потім ми опишемо вибрані методи, їх параметри та вибрані набори даних. Нарешті, ми будемо обговорювати постобробку та оцінку. Окремі деталі реалізації також обговорюються в цьому розділі, оскільки вони використовують вибір методу.

Основними завданнями для експериментальної частини буде реалізація детектора згорткових об'єктів, щоб перевірити, наскільки добре працює детектор, навчений загальними даними у конкретній задачі, а також потенційні методи для поліпшення.

Основними проблемами, що обмежують використання глибоких методів навчання, є нехватка продуктивності комп'ютера для навчання. Для цієї дисертації ми не мали доступу до серверної або високопродуктивного графічного процесора. Швидше, нам довелося реалізовувати методи на серійному ноутбукі. Навчання згорткової мережі від початку до кінця на такому обладнанні буде надзвичайно довгим. Таким чином, ми віддали перевагу таким методам, які були встановлені достатньо для того, щоб мати доступ до попередньо підготовлених мереж.

На додаток до обчислювальної потужності, глибокі згорткові мережі вимагають великої кількості навчальних даних. Оскільки збирання та анотування набору даних є витратним, більшість досліджень виконуються на декількох доступних наборах даних.

Загальний детектор об'єктів повинен бути навчений різним даним і підтримувати кілька категорій об'єктів. Стандартні набори тестів (PASCAL VOC та ImageNet) забезпечили відповідну відправну точку. Об'єктивний детектор об'єктів також повинен бути легко доступним і простим у



застосуванні. Існує безліч доступних мереж, які були попередньо підготовлені в стандартних наборах даних.

Для оцінки ефективності пошуку об'єктів за конкретною задачею ми зібрали дані тесту з різних джерел. Основний критерій полягав у тому, що об'єктові анотації даних тесту повинні бути приблизно сумісними з об'єктом анотацій стандартних тестів. Це означає, що тестові набори даних повинні мати спільні класи з стандартними орієнтирами та включати в себе анотації об'єктів як обмежувальні поля.

### 3.1.1. Пошук об'єкта

Був обраний швидкий R-CNN як основний метод для експериментів із виявлення об'єктів. Швидкий R-CNN вже добре зарекомендував себе і має реалізацію та підготовку мереж для декількох різних платформ. Навіть незважаючи на те, що передові методи, такі як швидкий R-CNN, забезпечують незначне підвищення точності, основним їх внеском є покращення швидкості. Оскільки оцінка часу виконання була в основному виведена за рамки цієї тези, ці методи мали б додаткову цінність для експериментів.

Вирішено пропустити початкове навчання та використовувати попередньо навчені мережі VGG-16. VGG-16 в даний час є одним з стандартів. Структуру VGG-16 зображено на рис. 3.1.

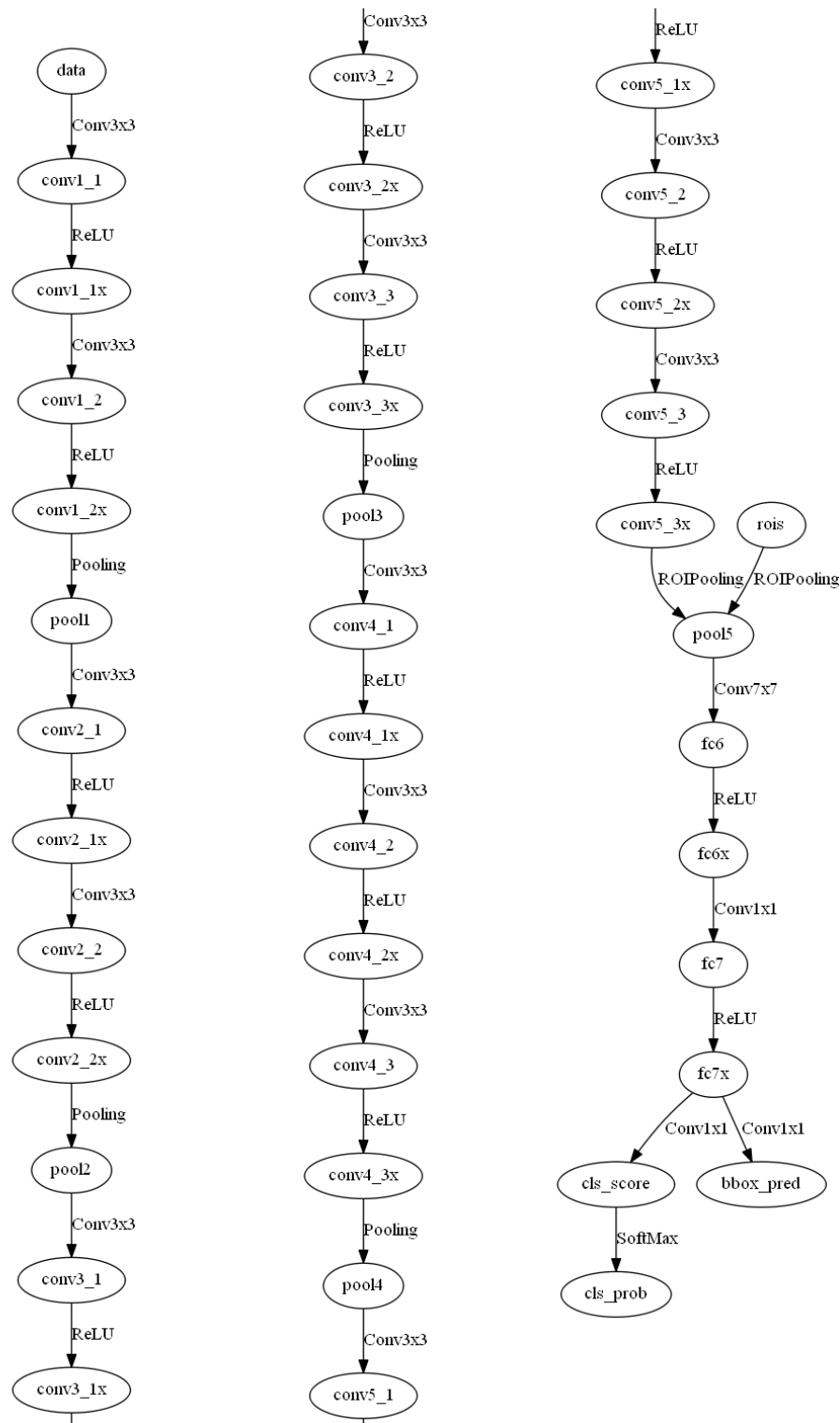


Рис. 3.1. Структура мережі Fast R-CNN VGG-16.

У оригінальній публікації швидкий R-CNN [1] VGG-16. Згадується як приклад "дуже глибокої мережі" і використовується як модель великої мережі у всіх експериментах. Було використано попередньо навчену мережу, структура якого показана на рис. 3.1. Більшість операцій мереж показані в вигляді окремих шарів. повністю згорткова підмережа, як показано початок в лівому

верхньому кутку. Після того, як кожні дві-три пари з них здійснюють об'єднання. Розміром кожного згорткового шару  $3 \times 3$  (визначено як розмір ядра розміром 3). Розмір ядра згрупованих шарів становить 2. Згорткові шари мають ступінь 1 і об'єднані шари мають ступінь 2.

The regions of interest - RoIs (Області зацікавленості). Ця мережа включає два повністю з'єднаних шари та два активаційні шари областей зацікавленості. Повністю з'єднані шари містять 4096 вихідних параметрів, які відображають розмір намальованої карти активації. Шар використовує відсів з співвідношенням 0,5. Після другого шару активації обробка відгалужується для обчислення вихідних ймовірностей та граничних значень поля.

Оскільки в якості детектора об'єктів був обраний швидкий R-CNN, також необхідні окремі методи виділення області. Оскільки їх було відносно легко реалізувати, ми вирішили експериментувати як з Selective Search, так і з Edge Boxes, які були описані в другому розділі.

Вибірковий пошук може використовувати кілька різноманітних способів. Додання різноманітних способів забезпечує кращі результати, але збільшує час обчислення. Оскільки час розрахунку не оцінюється. Було обрано режим налаштування, який автори називають "Вибірковий пошуку". Це включає в себе використання всіх чотирьох різноманітних способів та різних кольорових просторів: HSV, Lab, rgI (rg channels of normalized Інтенсивність RGB плюс), H (відтінок каналу від HSV) та I (лише інтенсивність).

Параметри методу генерації суперпікселів, які використовуються вибірковим пошуком, також внаслідок виходу. Параметр  $k$  визначає кількість використаних початкових сегментів. Автори вибіркового пошуку виявили, що алгоритм був точним для цілей детектування об'єкта з значеннями  $k$  між 50 і 300. Вибрали середнє значення  $k = 200$ . Перед сегментацією суперпікселів, зображення згладжується за допомогою Фільтра Гауса. Оригінальні автори [4] завжди використовують  $\sigma = 0.8$ . Які вони виявили корисними при видаленні артефактів цифровизації без змінення образу.

### 3.1.2. Об'єкти у перспективі

Видання 2008 \ Putting Objects in Perspective " [5] (OiP) Hoiem et al. Використовує сегментацію геометрії як допомогу для виявлення об'єктів. Основна ідея методу - змінити вірогідність виявленого об'єкта на основі того, наскільки геометрично ймовірно місце розташування об'єкта.

Метод використовує навчальні дані для оцінки ймовірностей різноманітних геометричних поверхонь (що складаються з поверхні об'єкта та сусідніх поверхонь) з урахуванням класів об'єктів. Метод також оцінює точку розташування камери. Оцінюється розташування камери на сцені та положення горизонту на площині фотографії. Виходячи з цих даних, можна підрахувати висоту об'єктів на зображенні. Якщо висоти об'єктів відомі, вірогідність екземплярів об'єкта може бути скоригована залежно від того, чи вони потрапляють в очікуваний діапазон висот. Також можна виявити, чи знаходиться об'єкт на земельній площині, і відповідно відрегулювати ймовірність.

Об'єднавши ці дані, включають велику кількість інформації про можливі місця розташування об'єктів. Детектор об'єкта підключений до цих оцінок. Враховуючи взаємодію між точкою зору, виявлення об'єктів та геометрією поверхні, враховуючи належний висновок, який обчислюється за допомогою алгоритму розповсюдження вірувань.

### 3.2. Набір даних

Стандартні орієнтири, що використовуються для виявлення об'єктів, є набір даних для завдання PASCAL (VOC) і набір даних для розпізнавання ImageNet у великих масштабах.

Основним набором тренінгів був пакет даних PASCAL VOC. У 2007 році набір даних містив 9,963 анотованих зображень з 20 класами об'єктів (людина, 6 різних тварин, у тому числі автомобілів). На наступний рік набір даних

збільшився до 54,9 тис. Зображень, близько половини з яких склали тестові та перевірочні набори. Кількість і тип різних завдань також різняться щороку.

Передчасна мережа PASOCAL VOC, яку була використана, сама по собі була навченої на наборі даних ImageNet. Починаючи з 2013 року, проблема включала завдання виявлення об'єктів, в якій в даний час містяться 200 повністю помічених категорій (включаючи ті ж самі категорії, що і PASCAL VOC). Навчальний комплект складається з 456 567 зображень, а набір для валідації складається з 21121 зображень.

### 3.2.1.Перевірка даних

Розпізнавання об'єктів, таких як пішохід та виявлення транспортних засобів, є популярними темами дослідження у комп'ютерному зору. Такі об'єкти анотовані в багатьох публічно доступних збірках даних. Це дало джерело даних для тестування загального детектора об'єктів. Автомобілі та люди також анотовані в тестових наборах даних, забезпечуючи сумісність.

Перший набір тестів, який ми використовували, є тим самим набором даних, який використовувався в публікації \ Putting Objects in Perspective (OIP) [5]. Набір містить 600 тестових зображень (з яких 422 включають допустимі об'єкти) та 60 вибраних зображень встановлених валідацій з бази даних LabelMe. Об'єкти, що входять до зображень, були позначені обмежувальними полями. Ми використали 60 зображень валідації як нашого першого тестового набору, до складу якого входять 126 легкових автомобілів та 84 пішоходів.



Рис. 3.2: Приклади зображень з набору даних OiP, об'єкти автомобіля позначені синім та, пішоходи позначені червоним кольором.

Є два класи об'єктів: автомобілі (у тому числі регулярні легкові автомобілі, а також фургони та вантажні автомобілі) та пішоходів (вертикальні особи). Набір даних "PASCAL VOC" містить однакові категорії. Транспортні засоби набору даних PASCAL включають (пасажирські) автомобілі та автобуси як окремі класи, але ніяких інших чотириколісних велосипедів. Проте, це розумна гіпотеза про те, що детектор, навчений цим двом класам, також навчиться розпізнавати більш загальний автомобільний клас набору даних OiP. Інший клас OiP, пішохід, суворо кажучи, є підкласом категорії людей PASCAL. Однак, розумно припустити, що більшість людей на зображенні вулиці є пішоходами.



Рис 3.3: Приклади зображень з набору даних Street Scenes. Об'єктами автомобіля, об'єкти автомобіля позначені синім та, пішоходи позначені червоним кольором.

Набір даних Street Scenes включає в себе 3547 зображень, які були марковані для 9 різних об'єктів (автомобіль, пішохід, велосипед, будівля, дерево, дорога, небо, тротуар та магазин). Було використали 60 зображень, обраних з цього набору, як наш другий набір тестів. Ця частина набору включає 131 машину та 51 пішохід.

Розташування об'єктів анотовано за допомогою багатокутників, які тісно оточують об'єкт. Полігони можуть бути перетворені в обмежувальну область з урахуванням мінімальних та максимальних значень для координат  $x$  і  $y$ . Клас автомобіля включає в себе всі моторизовані транспортні засоби (з більш ніж двома колесами) шириною більше 64 пікселів. Пішоходи включають осіб, які ходять або стоять і мають висоту більше 32 пікселів. Обидва об'єкти позначаються лише якщо їх видима частина складає 75%.

### 3.3. Реалізація

#### 3.3.1. Використане обладнання

Основним обладнанням для реалізації нейронної мережі був використаний комп'ютер з такими характеристиками:

- Процесор: Intel Core i5-2500k
  - Кількість ядер: 4
  - Базова частота: 3,30 GHz
- Відеокарта: NVIDIA GeForce GTX 560
- Оперативна пам'ять: HyperX DDR4-2133 8192MB
- Жорсткий диск: Western Digital Blue 1TB 7200rpm
- Операційна система: Windows 10.

Основне програмне забезпечення MATLAB 2014b.

### 3.3.2. MatConvNet та швидкий R-CNN

Було реалізовано згорткову мережу, використовуючи MatConvNet, яка є спеціально розробленою для цієї мети інструментарію MATLAB. Тим не менш, Mat-ConvNet забезпечив усі необхідні функціональні можливості, і його простіше встановити.

MatConvNet - це сукупність функцій MATLAB, які впроваджують різні конструктивні блоки мережі. Обгортка CNN об'єднує ці блоки в повну мережу.

Обгортка DagNN створює об'єкт MATLAB, який включає функції, що забезпечують доступ до навчання та оцінки мережі та параметрів, що зберігає структуру мережі та ваги. Об'єкт може зберігатися на диску, що полегшує обмін попередньо навчених мереж.

Веб-сайт MatConvNet забезпечує попередньо навчені мережі, які були перетворені у формат MatConvNet з оригінальних реалізацій і відповідають оригіналам до певної числової точності. Ми використали реалізацію DagNN VGG-16 швидкий R-CNN, яка була попередньо підготовлена до набору даних PASCAL VOC 2007. Мережа також використовує дані Imagenet Challenge 2012, оскільки мережа Fast R-CNN була ініціалізована за допомогою мережі, навченої на даних ImageNet.



Був створений скрипт MATLAB для виклику функцій CNN та для виконання попередньої обробки, які не надаються MatConvNet. Сценарій починається, завантажуючи мережу. Мережа вводиться в тестовий режим, що вимикає навчання. Граничні поля потім перетворюються в формат координат, який використовується мережею. Далі зображення та коробки масштабуються відповідно до розміру входу в мережі (600x800 пікселів з 3 кольоровими каналами). Зображення також попередньо обробляються шляхом віднімання середнього значення кольору.

Після підготовки даних функція оцінки мережі викликається з використанням зображення та області як вхідних аргументів. Після оцінки, класові ймовірності та дельта-значення для кожної області знімаються "поза мережею". Далі рецензії обмежувальної копії обчислюються шляхом об'єднання вихідних значень та значень дельти. а також ймовірності(автомобілі, автобуси та особи) зберігаються в структурі. Замовлення автомобілів і автобусів поєднуються в оцінці для створення наближення більш загальноприйнятого класу автомобілів, який включає в себе великі автомобілі, а також легкові автомобілі. Виявлення зберігаються базі даних.

### 3.3.3.Геометричний висновок

MATLAB реалізація \ OIP та "Геометричний контекст" доступні в [8]. Пакет Ge-ometric Context включає в себе реалізацію C ++ методу генерації суперпікселів, для реалізації OIP потрібні обидва пакети. OIP пакет включає в себе модифіковану версію спливаючого коду Automatic Photo, який замінює функцію обробки каталогу пакета "Геометричний контекст". Ця функція вводить каталог зображень, суперпікселів, обчислених для зазначених ліній і попередньо навчених класів для сегментів зображення. сегментує зображення в каталозі та зберігає їх у вихідному каталозі.

Фактичний геометричний висновок був реалізований шляхом зміни сценарію, включений в OIP пакет. Сценарій спочатку завантажує сегментацію

зображення з диска і зберігає їх у структурі. Далі завантажуються ймовірності різних геометричних сегментів (заданих об'єктом). Вони надаються в пакеті для автомобілів і пішоходів. Тоді точки зору апріорні ініціалізуються. Для них ми використовували надані за замовчуванням розподіли (найімовірніше, висота камери 1.50 м, і, найімовірніше, розташування горизонту знаходиться посередині зображення).

Метод ОіР сподівається, що на виявлення не було здійснено максимального придушення. Скоріше, як частина конверсії виявлення перехресні виявлення згруповані разом у комбіноване виявлення, яке спочатку має таку саму ймовірність, що і найвищий коефіцієнт обмеження у групі. Вибір обмежувальної області може змінюватися як частина геометричного висновку.

Тоді ми називаємо функцію висновку, надану в пакеті ОіР. Функція приймає в якості вхідних виявлення у форматі кандидатів, сегментаціях зображення та пріоритети точки зору. Функція виводить нових кандидатів і розподіли ймовірностей розташування об'єкта. Ми знову конвертуємо нових кандидатів у раніше використаний формат виявлення та зберігаємо результати на диску.

### 3.4. Оцінка

#### 3.4.1. Швидкий R-CNN

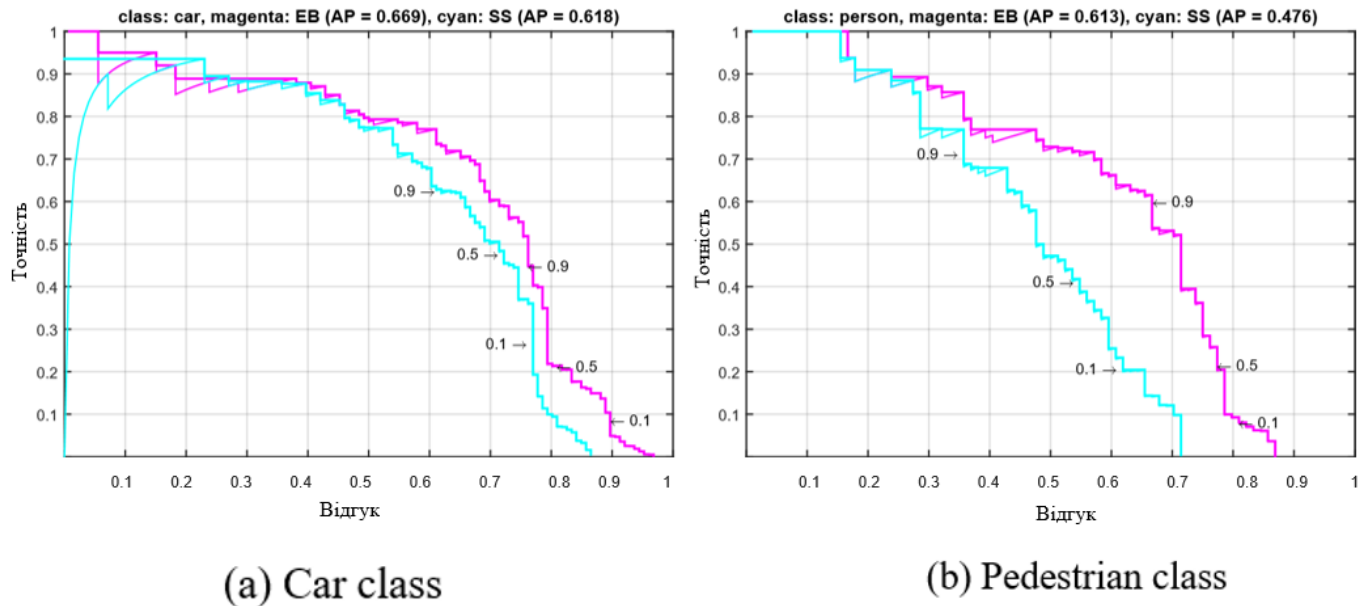


Рис. 3.4. Криві точності вилучення швидкостей R-CNN-виявлення даних ОіР з використанням селективного пошуку (блакитним) та граничних областей (пурпурним) як методів генерації регіону

На рис. 3.4 показані криві точності нагадування та середні точні дані швидких виявлень R-CNN набору даних ОіР. Виявлення були обчислено використовуючи два інші методи генерації: Selective Search (блакитний) і Edge Boxes (пурпурний). Як ви бачете, Edge Boxes має кращі результати (збільшилася середнє значення на 0,0510 для автомобілів і 0,1370 для пішоходів). Зауважте, що обидва способи використовують попередньо вибрані параметри. Цілком можливо, що тонке налаштування параметрів могли б забезпечити кращі результати за обидва методи.

Загалом, результати виявлення Fast R-CNN досить непогані, що згортована мережа навчалася на зовсім іншому наборі даних, який не включає в себе багато навчальних зображень, пов'язаних із автомобілями. Для автомобілів, до 40% правильних випадків викликаються з 90% точністю. Якщо метод використовуватиметься у практичному застосуванні, нам потрібно буде вибрати певну порогову вагу, над якою ми розглядаємо знайдені об'єкти з вищою вірогідністю. Три приклади вагових розташувань (0,1, 0,5, 0,9) відображаються поруч з кривими. Як ми можемо бачити, Edge Boxes забезпечує

більше результатів у кожному випадку, але меншою точністю. Це, мабуть, пояснюється тим, що Edge Boxes повертають багато кандидатів (в середньому, 8000 областей для зображень OiP, у порівнянні з 1000 областей, які повертаються шляхом вибіркового пошуку).

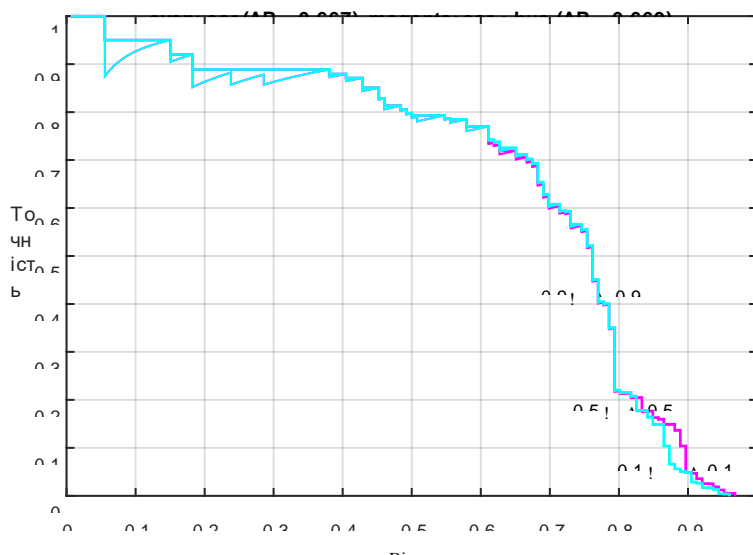


Рис 3.5: Додавання класу автобус до машин

Як було пояснено вище, виявлення для автомобілів є комбінацією класів автомобілів і автобусів PASCAL VOC, оскільки автомобілі визначаються більш широко в тестових наборах даних. Ми також виконували ті ж експерименти, використовуючи тільки клас машини PASCAL VOC. Це призвело до незначного зниження середньої точності (0.667 проти 0.669). Відповідні криві точності нагадування відображені на рис. 3.5. Ми бачимо, що детектор автобуса має вплив лише в кінці кривої. Тим не менше, відмінності показують, що відмінності між легковими автомобілями та автобусами розуміються детектором об'єктів досить істотні.

### 3.4.2. Позитивні помилки та негативні помилки



Рис. 3.6: Помилкові виявлення людей з набору даних OiP.

Рис. 3.6 показує помилково виявлених людей з набору даних OiP (використовуючи Edge Boxes і Швидко R-CNN). За візуальним оглядом близько на 20 містяться люди. Решта схожи на них, знаки дорожнього руху, конуси та невідомі об'єкти. Очевидно, Fast R-CNN виконує значно краще, ніж показує крива точності. Більшість з них - пішоходи, які частково закриваються або стоять занадто близько до інших людей. Посторонні об'єкти, мабуть, пов'язано з відсутністю жорстких негативних прикладів.



Рис. 3.7: Негативні випадки з набору даних OiP.

Відповідні помилкові виявлення відображені на рис. 3.7. Це пішоходи, яких не знаходив Fast R-CNN. Деякі випадки - люди, що стоять у тіні. Ці зразки

важко виявляти з повного зображення навіть людьми. Кілька інших - це високі контрастні предмети, що стоять перед аналогічним кольоровим фоном.

### 3.4.3. Не максимальна придушення

На рис. 3.8 показано ефект налагодження строгості не максимального придушення. Криві відображають середню точність виявлення автомобіля за даними OIP. На правому кінці кривої, де  $\text{IoU} = 1$ , ми взагалі не виконуємо NMS (за винятком виключення точних дублікатів, які вже мали відбутися). Ліворуч на кривій, параметр  $\text{IoU}$  зменшується, що означає, що ми починаємо видаляти виявлення з зменшенням перекриття.

Ми бачимо, що Edge Boxes вигідніший від NMS з "обох сторін", оскільки як суворе, так і м'яке придушення погіршують результати. З іншого боку, Selective Search показує поліпшення, починаючи з приблизно  $\text{IoU} = 0.6$ , але після цього зменшення значення параметра не забезпечує суттєвих змін точності. Це, мабуть, через виборчий пошуку, який видає менше результатів і гірше працює. При  $\text{IoU} = 0.6$ .

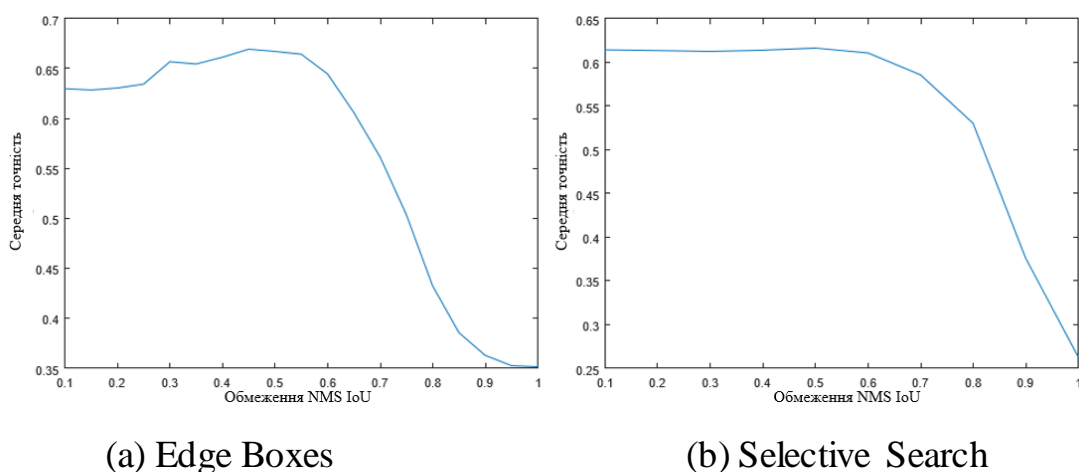


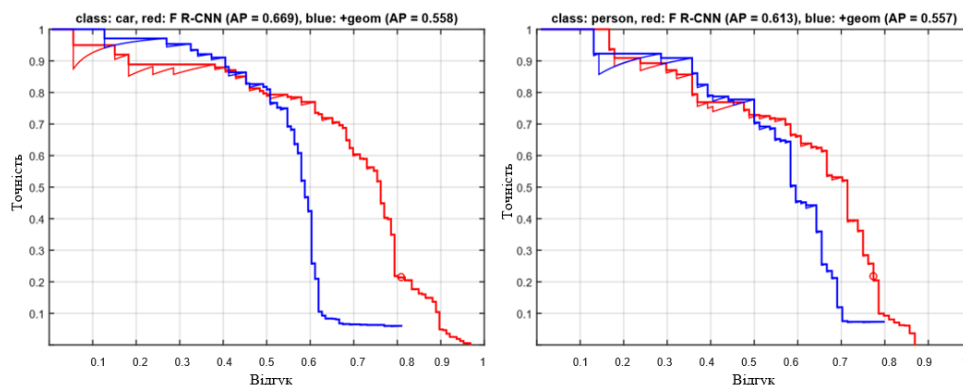
Рис. 3.8: Ефект зміни параметра NMS IoU на середню точність швидких виявлень R-CNN.

Вторинні виявлення були видалені. Знижуючи межу, теоретично, NMS в якийсь момент має почати видаляти суперечливі дійсні позитивні виявлення та зниження точності. Проте в результатах вибіркового пошуку ці бокси з якоїсь причини відсутні.

Результати для виявлення особи були подібними. Виходячи з цих результатів,  $IoU = 0.45$  було вибрано як оптимальне значення NMS для експериментів.

#### 3.4.4. Геометричний висновок

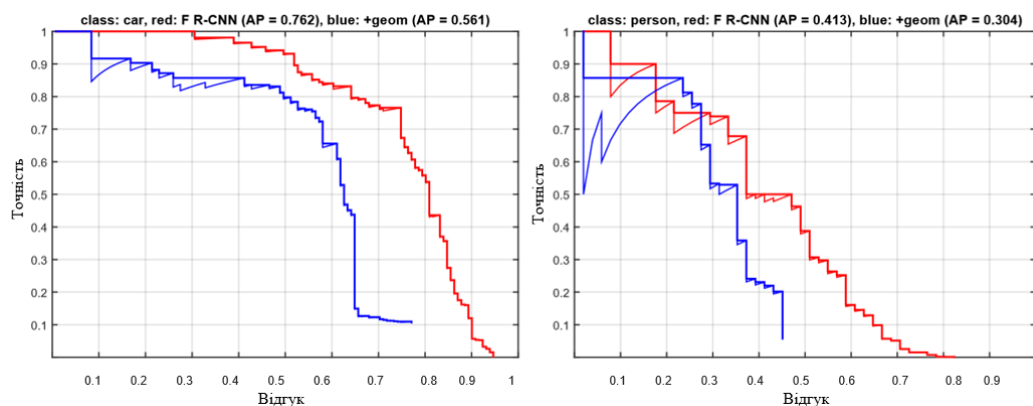
На рис. 3.9 та рис. 3.10 показано вплив описаного геометричного методу виведення на виявлення об'єктів. На рис. 3.9 показана крива точності виклику Fast R-CNN (червоний) даних OiP у порівнянні з кривою тих же результатів після геометричного виведення (синього кольору). рис. 3.9 показує подібне порівняння даних.



(a) Автомобіль

(b) Персона

Рис. 3.9: Порівняння Fast R-CNN (червоний) та OiP + Fast R-CNN (синій) за допомогою набору даних OiP



(a) Автомобіль

(b) Персона

Рис. 3.10: Порівняння Fast R-CNN (червоний) і OiP + Fast R-CNN (синій), використовуючи набір даних "Вуличні сцени"

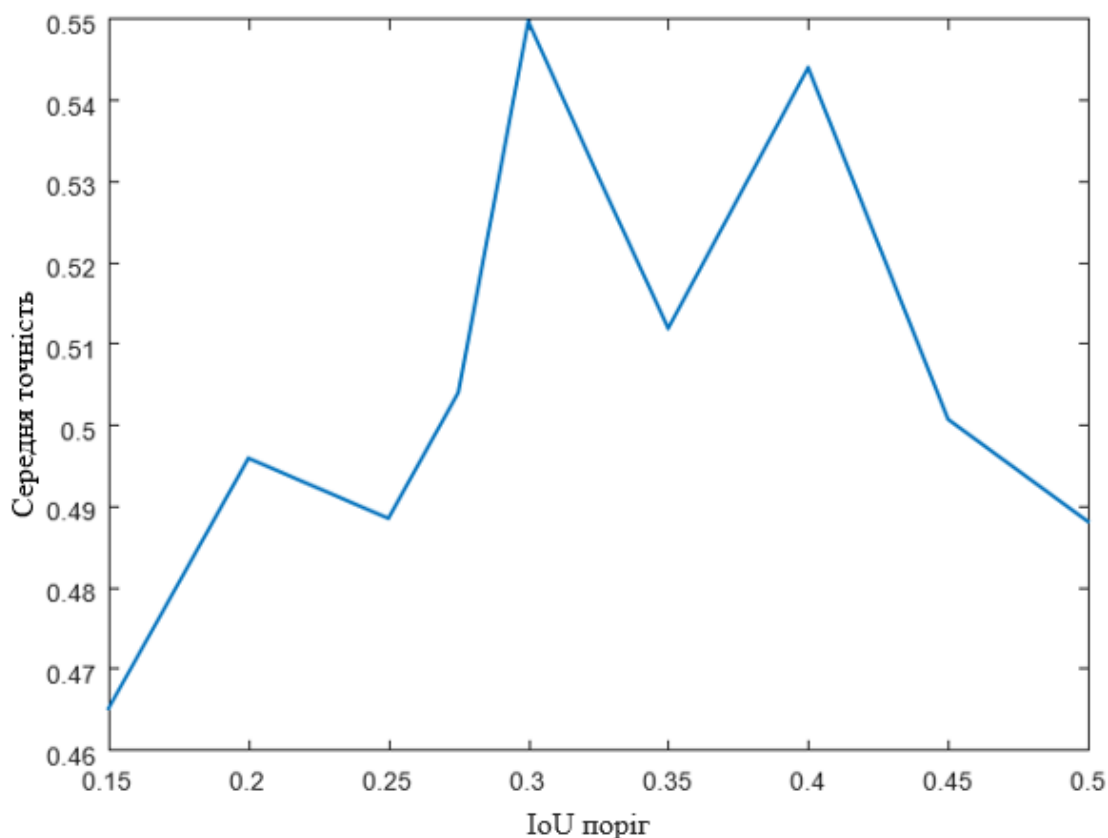


Рис. 3.11: Ефект зміни параметра порогового значення IoU групування на середню точність після геометричного висновку.

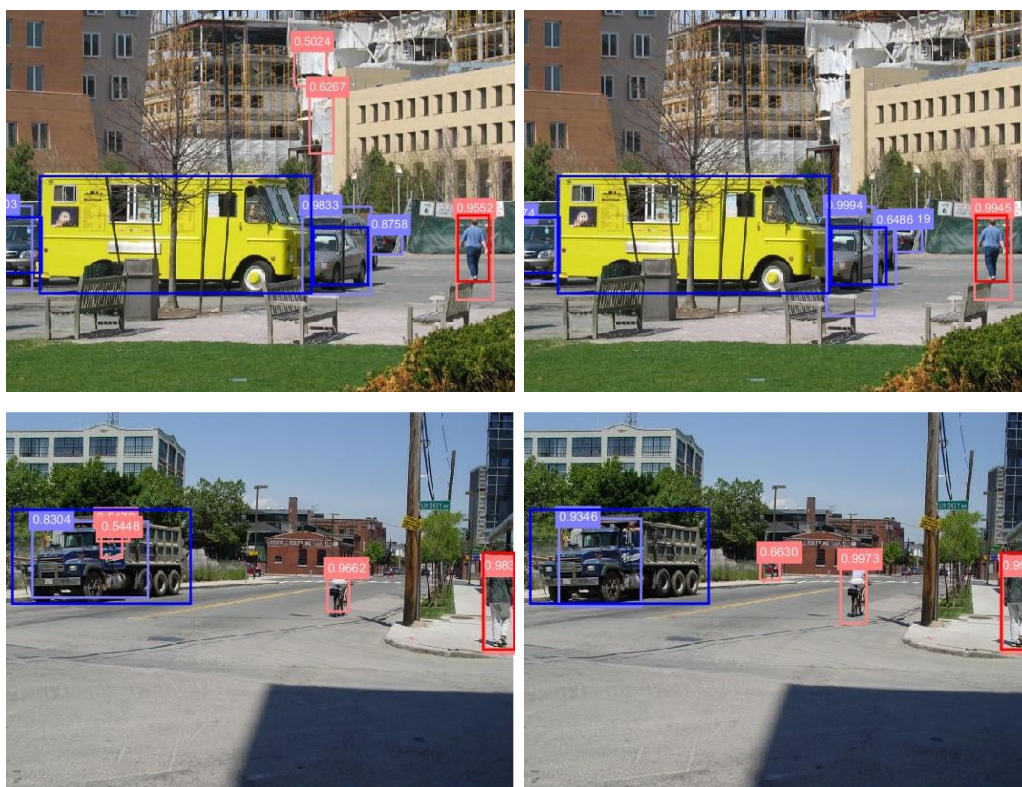
Як зазначено вище, виявлення Fast R-CNN були після обробки за допомогою NMS з порогом  $\text{IoU} = 0.45$ . Оскільки метод геометричного виведення не використовує звичайні NMS, то не має сенсу проводити порівняння з використанням одного і того ж порогового значення для обох методів. Замість цього ми обрали оптимальні порогові значення для NMS та методу групування окремо та порівнювали результати. Рис. 3.10 показує, як виявляє людину геометричний метод, коли змінюється поріг IoU. Виходячи з цих результатів, ми вибрали 0,3 як оптимальний пороговий розподіл, а криві з точністю відкликання були намальовані за допомогою цього значення. З міркувань продуктивності швидкі виявлення R-CNN з ймовірністю до 0,01 були відфільтровані перед геометричним виведенням. Це пояснює, чому крива трохи змінюється перед швидкою R-CNN кривою.

Криві точності ясно показують, що, принаймні на цих наборах даних та анотаціях, геометричний висновок значною мірою знижує продуктивність Fast



R-CNN. Виявлення автомобіля набору даних OiP є єдиним випадком, коли синя крива спочатку перевищує червону криву, що означає, більш надійні результати після геометричного висновку.

Однак візуальним оглядом окремих результатів ми можемо чітко знайти приклади роботи геометричного висновку, як це передбачено. Деякі приклади виявлення показані на рис. 3.12. Тут виявлення проявляються лише тоді, коли їх ймовірність перевищує 0,5. У верхньому зображенні ми бачимо, як геометричний висновок правильно знижує оцінку реклам (що висять з боку будівлі). Деякі автомобілі та єдиний пішохід правильно виявляються обома способами. Жовтий автомобіль не визначається жодним способом, ймовірно, тому що, будучи закритими іншими об'єктами, воно не виявляється в першу чергу як об'єкт-кандидат.



(a) Швидкий R-CNN

(b) Швидкий R-CNN + геометрична модель

Рис. 3.12: Приклади виявлення. Особа класу позначена червоним, а клас автомобіля позначений синім кольором.

Проте геометрична модель також дає неповні виявлення автомобілів, які не ановані в оригінальних даних (насправді, не так багато автомобілів на

зображенні). Це ілюструє характерну поведінку геометричного методу. Швидкий R-CNN вже досить добре працює у визначенні класів об'єктів, але через оклюзії та неповних анотацій є багато помірно можливих часткових виявлень, які не враховуються як справді позитивні при оцінці. Якщо ці виявлення також є геометрично правдоподібними, метод геометричного виведення збільшує їх вірогідність ще більше. Отже, якщо метод правильно зменшує кількість хибно-позитивних виявлень, водночас він пропагує багато інших, які є геометрично в правильному місці, але фактично є неправильними (наприклад, знаки дорожнього руху та повідомлення про стан води, помилкові). Наявність великих груп людей та автомобілів часто призводить до виявлення, що є частковими (оскільки вони закриті), а не помічені в вихідних даних, і їх сприяють геометрична модель.

Метод геометричного виведення не покращує середні результати Fast R-CNN за розміченими об'єктами, але в деяких випадках, метод вказує на кращі результати, які можуть бути застосовані як у виявленні згорткових об'єктів, так і в геометричному висновку.

### 3.5. Android додаток

#### 3.5.1. Клієнтський інтерфейс Android додатку

Клієнтський інтерфейс можна розділити на п'ять частин:

- Вікно авторизації користувачів

Тут здійснюється авторизація користувачів для введення статистики та для збереження їхніх результатів в захищеній хмарі.

На рис. 3.13 зображено вигляд цього вікна.

Авторизація буде проходити в три варіанти:

- Рестрація та вторизація нового користувача
- Авторизація за допомогою соціальної мережі Facebook
- Торизаця за допомогою акаунту Google
- Вікно активної камери

Тут буде проводитися серія знімків, бажано здійснювати цю серію з штативу так як рображення будуть рівнішими і збільшиться точність видалення не потрібних об'єктів.

Також в майбутньому буде підтримка фільтрів, так як це є популярним для сучасних користувачів соціальних мереж.

На рис. 3.14 зображено вигляд цього вікна.

- Вікно отриманого результату

Тут зроблена серія знімків буде відправлена на сервер де вище описана нейронна мережа знайдене потрібні об'єкти та окремий алгоритм видалить їх з фотографії.

На рис. 3.15 зображено вигляд цього вікна.

- Вікно історії здійснених фотографій

Тут будуть показані зроблені в минулому серії знімків та результати їх обробки на сервері.

На рис. 3.16 зображено вигляд цього вікна.

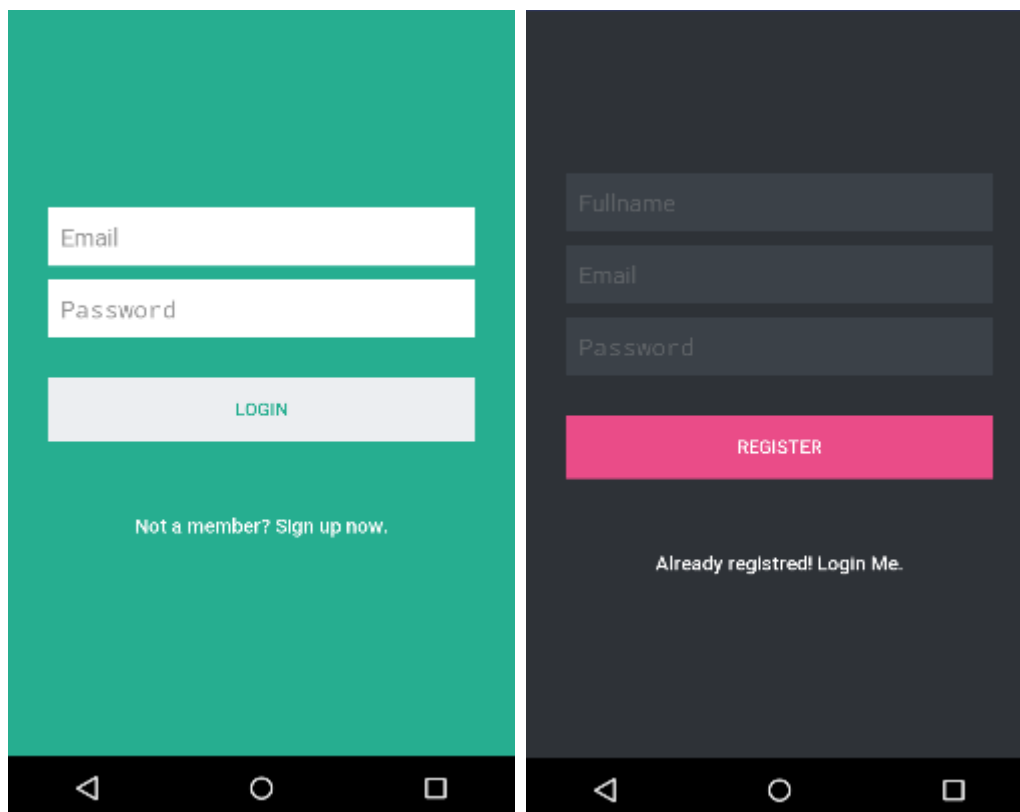


Рис. 3.13 Вікно авторизації та реєстрації користувачів



Рис. 3.14 Вікно активної камери

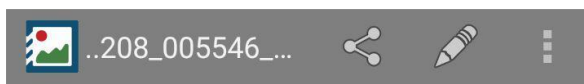


Рис. 3.15 Вікно отриманого результату

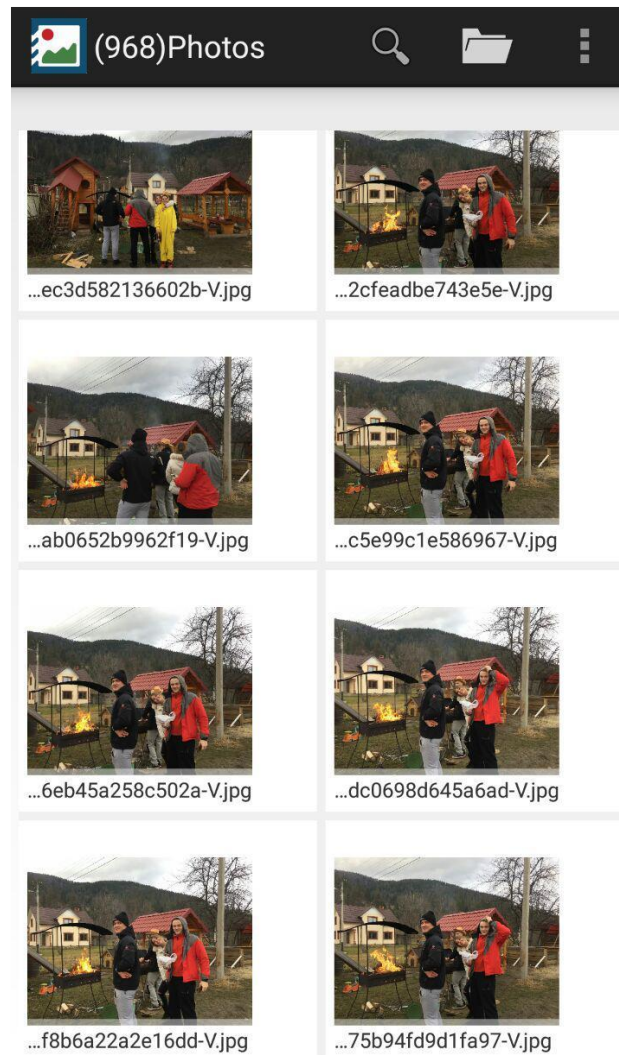


Рис. 3.16 Вікно історії здійснених фотографій

### 3.5.2. Структура роботи андроїд додатку

На рис. 3.17 зображено загальну UML діаграму менеджера фотографій, тут показано вікна та їх взаємодію між собою. На рис.3.16 показано UML загальну діаграму камери.

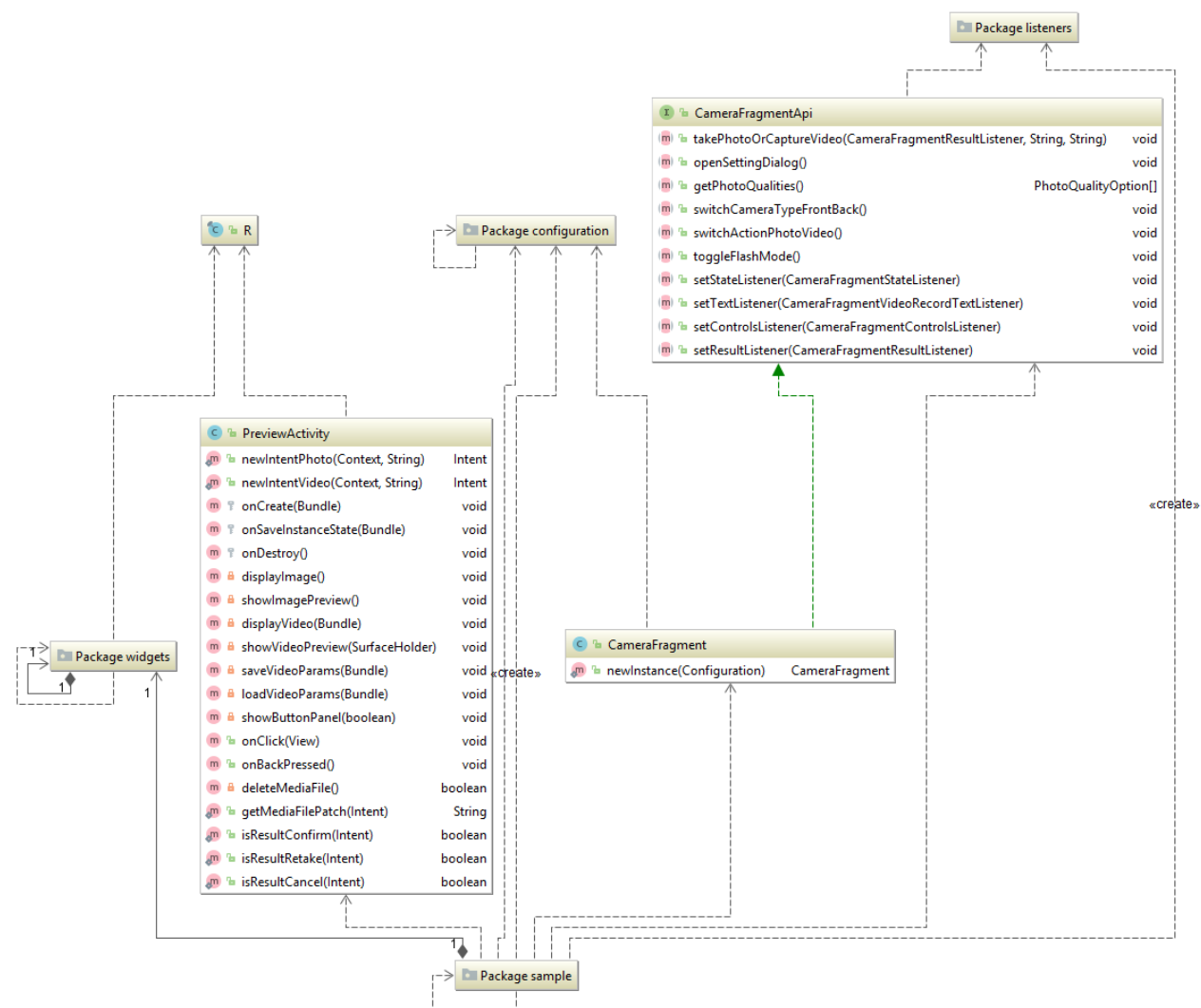


Рис. 3.17 Загальна UML діаграма камери

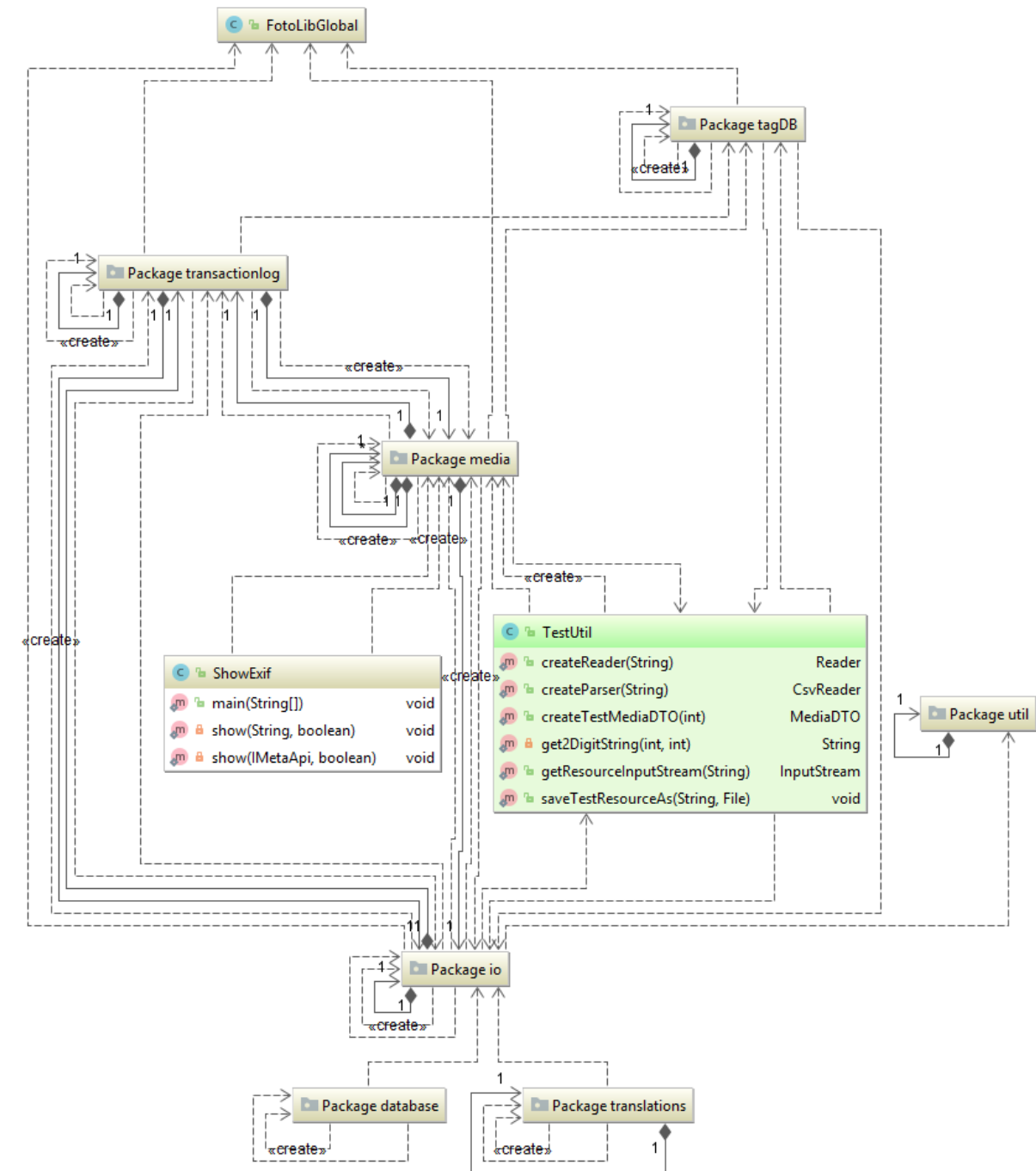


Рис. 3.18 Загальна UML діаграма менеджера фотографій

### 3.5.3. Структура роботи серверної частини

Вся важкість системи це робота нейронної мережі. На рис. 3.19 зображено вище описану роботу нейронну мережу.



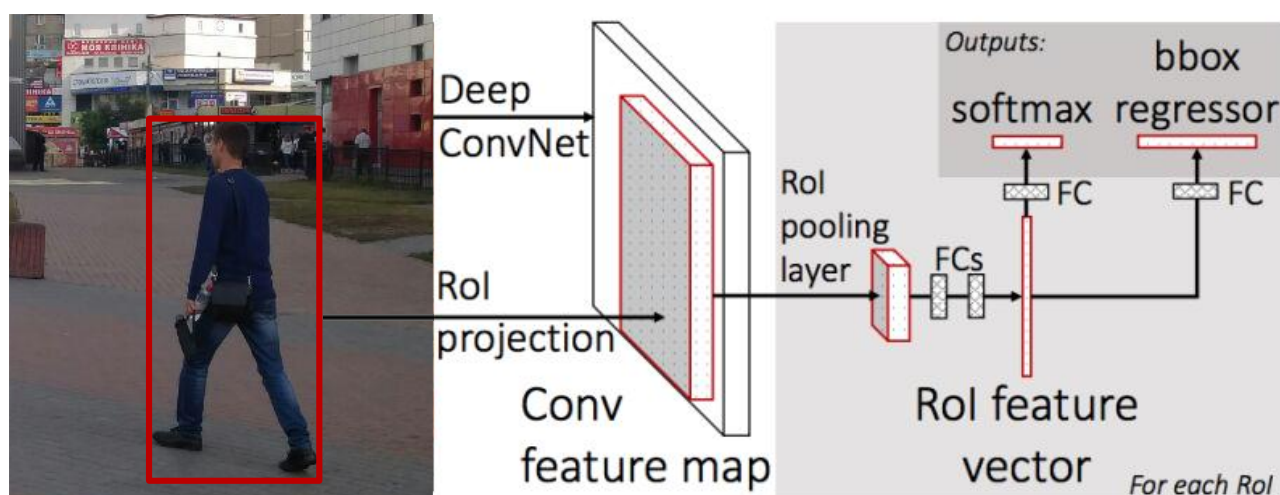


Рис. 3.19 Схема роботи нейронної мережі

### Висновки по розділу

На практиці, для експериментування методом згортки, ми створили робочу програму MATLAB Fast R-CNN. Ми дізналися, що найбільш складною частиною впровадження глибокої системи навчання є збір даних для навчання та проведення самого навчання. Доступні набори даних є корисною відправною точкою як для дослідження, так і для практичних реалій. Тривалість навчання може бути ще скорочена за допомогою попередньо навченої мережі. Навіть якщо кінцева система не має тих же класів об'єктів, що і еталонні дані, візуальні проблеми є достатньо універсальними, аби отримати вигоду від детекторів, навчених для іншої проблеми. Оптимальні нижні шари згорткової мережі часто схожі, незалежно від проблеми, так само, як людське око використовує ті ж поля для всіх візуальних завдань. Таким чином, має сенс ініціалізувати шари за допомогою попередньо навченої мережі.

Що стосується реалізації, що не існує простих "нестандартних" рішень для ефективного впровадження згорткових мереж. Поточні програмні засоби, такі як Caffe та MatConvNet, вимагають спеціальних навичок. Якщо можна використовувати такі інструменти, створення робочої реалізації не є надто складним. Тим не менше, інструменти досить вигадливі щодо сумісності версій програмного забезпечення та апаратного забезпечення. Що стосується android



додатку був розроблений простий в користуванні інтерфейс який об'єднує людей з такою неймовірною системою як нейронна мережа, в якій волищезний потенціал в різних сферах.

## Розділ 4. МАРКЕТИНГОВИЙ АНАЛІЗ СТАРТАП-ПРОЕКТУ

### 4.1.Опис ідеї проекту

Таблиця 4.1. Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Робочий додаток на мобільний пристрій завдяки якому є можливість фотографувати об'єкти або різні пейзажі при цьому будуть видалятися рухомі об'єкти	Туризм	Туристи можуть фотографувати об'єкти або різні пейзажі при цьому будуть видалятися рухомі об'єкти.
	Комерційні потреби	Використання в комерційних потребах це здійснення фотографій для реклами та швидко її опубліковувати

Таблиця 4.2. Опис ідеї стартап-проекту

№	Техніко-економічні характеристики ідеї	Продукція конкурентів			W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Дипломний проект	Remove	Adobe Photoshop Fix			
1	Швидкість роботи	Швидка	Середня	Повільна	Зв'язок з мережею	Простота архітектури	Потрібно менше знань для виконання заданої задачі
2	Зручність використання	Зручно	Зручно	Зручно	Для здійснення серії знімків потрібно використовувати шпатель	Розвиток UI має перспективу через наявність API	Автоматизована робота
3	Вимоги до системи	Мінімальні	Мінімальні	Високі	Відсутня оптимізація зі старими системами	Простота системи	Актуальність програми для нових систем через використання новітніх бібліотек

Закінчення таблиці 4.2

4	Кросплатформність	Відсутня	Відсутня	Наявна	Відсутня оптимізація для роботи через мобільні пристрої	Налаштована система для роботи через головні інструменти і використання користувачами	-
---	-------------------	----------	----------	--------	---	---	---

#### 4.2. Технологічний аудит ідеї проекту

Таблиця 4.3. Технологічна здійсненність ідеї проекту

№	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Видалення рухомих об'єктів з серії знімків	Використання згорткової нейронної мережі	Fast R-CNN	Повністю відкрита для роботи з інформацією
2	Робота безпосередньо з інформацією, яку надає користувач	API для роботи з обробки зображень	Розроблені бібліотеки для роботи нейронної мережі	Повністю відкритий програмний код та використання готових бібліотек
Обрана технологія реалізації ідеї проекту:1				

Висновок: технологічна реалізація продукту – можлива, вибрана технологія №1 яка може нам допомогти розробити якісний продукт з використанням комбінації технологій, та перспектива у майбутніх ідей.

#### 4.3. Аналіз ринкових можливостей запуску стартап-проекту

Таблиця 4.4. Попередня характеристика потенційного ринку

№	Показники стану ринку	Характеристика
1	Кількість головних гравців, од	4
2	Загальний обсяг продаж, грн./ум.од	200
3	Динаміка ринку	Темпи розвитку світової економіки позитивні, але з ознаками зменшення росту.
4	Наявність обмежень для входу	Відсутні. Так як аналоги використовують не автоматизовані алгоритми обробки фотографій
5	Специфічні вимоги до стандартизації та сертифікації	Відсутні.
6	Середня норма рентабельності в галузі або по ринку, %	70

Висновок: враховуючи кількість головних гравців по ринку, зростаючу динаміку ринку, невелику кількість конкурентів та середню норму рентабельності можна зробити висновок, що на даний момент, ринок для входження стартап-продукту є привабливим.

Таблиця 4.5. Характеристика потенційних клієнтів стартап-проекту

№	Потреба, що формує ринок	Цільова аудиторія	Відмінності у поведінці цільових груп клієнтів	Вимоги споживачів до товару
1	Потреба в зменшенні витрат в часі та батареї телефону	Туристи	Стартап буде переважно буде економити велику кількість часу та витрат заряду батареї	Зручність у використанні. Швидка робота системи. Спроможність швидко освоїти як користуватися системою. Можливість редагувати не вірні результати роботи нейронної мережі.
2	Потреба в зменшенні витрат в часі та швидкої публікації фотографій	Звичайні користувачі	Стартап буде переважно буде економити кількість часу та швидкої публікації фотографій в соц мережі	

Таблиця 4.6. Фактори загроз

№	Фактор	Зміст загрози	Можлива реакція компанії
1	Конкуренти	Наявність конкурентів котрі надають схожі рішення	Зменшення ціни на поставлену послугу; Розробка унікальних характеристик товару; Надання ліцензій на обслуговування
2	Кошти на розробку та підтримку продукту	Закінчення грошей та недостатнє фінансування	Залучення додаткових інвесторів, мотивація роботи на перспективу; Ітеративна розробка продукту задля покрокового виведення продукту на ринок та отримання відповіді користувачів
3	Вихід аналогу	Вихід аналогу даного товару може призвести до знецінення та безідейності даного товару	Вихід товару на ринок в коротші строки з не повною, але достатньою, функціональністю для зацікавлення усіх цільових аудиторій; Проведення рекламної компанії

Таблиця 4.7. Фактори можливостей

№	Фактор	Зміст можливості	Можлива реакція компанії
1	Новий продукт	Вихід на ринок, Зменшення монополії, Надання нових рішень у сфері	Розробка нової функціональності; Вихід нової продукції на ринок; Надання різноманітних типів ліцензій в залежності від потреб користувача \ замовника.
2	Вихід аналогу	Надати продукт з певними характеристиками та можливостями що відсутні у компаній конкурентів	Аналіз ринку та користувачів задля задоволення їх потреб та надання функціональності у найкоротші строки за ціну, котра є дешевшою ніж у продуктів-замінників.

Закінчення таблиці 4.7

3	Зворотній зв'язок від користувачів	Можливість отримання необхідної інформації для вдосконалення продукту	Наявність вхідних даних та реакція на них з боку команди розробників задля задоволення потреб та бажань кінцевих користувачів системи кешування даних.
4	Грошова винагорода за рекламу	При достатньому попиті на систему кешування даних можлива комерціалізація продукту на основі реклами задля отримання грошової винагороди для подальшого розвитку продукту та оплати заробітної плати працівникам	Точкова комерціалізація продукту; Введення реклами; Ведення додаткових коштів у проект задля його подальшого розвитку.

Таблиця 4.9. Ступеневий аналіз конкуренції на ринку

№	Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1	Тип конкуренції: монополістична	Товар від кожної компанії на ринку, являється недосконалим замінником товару, реалізованого іншими фірмами; На ринку є умови для входу та виходу; Ціна корелює між суперниками;	Розробка продукту з характеристиками, які покривають сфери вживання що не покривають інші товари-замінники; Кореляція цін у відповідності до товарів замінників; Різні типи ліцензій.
2	Рівень конкурентної боротьби: світовий	Всі продукти замінники розроблялись інтернаціональними командами з різних куточків світу, продукти не належать до певної держави, а належать команді розробників	Вихід на ринок збуту продукту з клієнто-необхідною функціональністю; Налагодження маркетингу на основних Інтернет ресурсах задля охоплення великої кількості потенційних користувачів; Надання бета-версій продукту.

Закінчення таблиці 4.9

3	Галузева ознака: внутрішньогалузева	Даний тип продукту може використовуватися тільки у сфері розробки ІТ додатків \ продуктів	Надання зручного, інтуїтивно зрозумілого інтерфейсу; Підтримка всім відомих методів взаємодії з середовищем розробки; Наявність документації та он-лайн підтримки.
4	Конкуренція за видами товарів: товарно-видова	Дана конкуренція – конкуренція між товарами одного виду.	Впровадження функціональності яка відсутня у товарів-замінників; Спрощення інтерфейсів; Надання підтримки.
5	Характер конкурентних переваг: цінова та не цінова	Цінові переваги – точкова комерціалізація; Не цінова – надання функціональності, що відсутня у товарах- замінниках.	Надання платних ліцензій лише на критично важливу функціональність для клієнта з певним строком підтримки, що зазначена у відповідній ліцензії; Впровадження унікальної функціональності.
6	За інтенсивністю: марочна	Наявність унікального знаку що відрізняє даний продукт від продуктів-замінників	Впровадження власної назви та власного знаку.

Таблиця 4.10. Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари- замінники
	ADOBE	Scalado	Google Play, Apple Store	Форуми, точки продажу додатку	Продукція компанії ADOBE
Висновк и	Прямі конкуренти намагаються сконцентруватис я на інших напрямках свої продуктів.	На даний момент їх додаток який може дати нам конкуренці ю не дорівнює і не впроваджен ий.	Постачальники диктують умови збереження даних, які захищають приватність. Також постачальники не дають змогу зловживати нормами.	Клієнти можуть диктувати умови на ринку через повідомленн я на форумах або в полі відгуків в точках продажу додатку.	Можливість введення стандартизації ї роботи алгоритмів розпізнаванн я рухомих об'єктів

Проаналізувавши можливості роботи на ринку з огляду на конкурентну ситуацію можна зробити висновок: оскільки кожний з існуючих продуктів не впливає у великій мірі на поточну ситуацію на ринку в цілому, кожний з існуючих продуктів має свою специфічну сферу використання та свої позитивні та негативні сторони щодо рішення певних типів задач, то робота та вихід на даний ринок є можливою і реалізованою задачею.

Для виходу на ринок продукт повинен мати функціонал що відсутній у продуктів-аналогів, повинен задовольняти потреби користувачів, мати необхідний та достатній функціонал з конфігурування, підтримку зі сторони розробників та можливість розробки спеціального функціоналу за відповідною ліцензією.

Таблиця 4.11. Обґрунтування факторів конкурентоспроможності

№	Фактор конкурентоспроможності	Обґрунтування
1	Прагматичність	Через запуск стартапу система буде не дуже складно з точки зору архітектури перший час. Через певний період із додаванням функціоналу та оптимізації алгоритмів роботи програмний код буде все складнішим. Такий етап наступить не раніше одного року постійної роботи над проектом.
2	Зручність	Оскільки стартап розробляється на багатьох платформах з різною шириною екранів, то зручність використання системи на різних пристроях буде відігравати не малу роль у спроможності конкурувати з іншими гравцями ринку.
3	Швидкість роботи	Швидкість роботи відіграє велику роль для користувачів, оскільки вони не будуть готові чекати декілька хвилин на виведення результату роботи додатку.
4	Оптимізація	Якщо додаток буде дуже часто видавати помилки при роботі, то користувачі не будуть вважати додаток надійним.



Закінчення таблиці 4.11

5	Налаштування під користувача	Різні люди мають різні звички, які вони використовують, наприклад, якщо є люди, які люблять працювати за додатком де є темні кольори, а є такі люди, які люблять світлі кольори. Можливість редагувати зовнішній вигляд додатку надає значну перевагу серед конкурентів.
6	Приватність	В останні роки приватність людей та інформація щодо них все частіше зловживається шахраями або великими корпораціями, які потребують погодження з умовами доступу до приватної інформації та її обробки..
7	Технічна підтримка	Якщо технічна підтримка компанії буде працювати своєчасно та швидко, то це допоможе зберегти репутацію компанії на відміну від конкурентів, де їй не приділяють увагу.

Таблиця 4.12. Порівняльний аналіз сильних та слабких сторін системи кешування  
мало змінних даних

№	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з запропонованим						
			-3	-2	-1	0	+1	+2	+3
1	Прагматичність	12	Remove	Adobe Photoshop Fix	-	0	-	-	-
2	Зручність	10	-	Remove	-	0	Adobe Photoshop Fix	-	-
3	Швидкість роботи	14	Adobe Photoshop Fix	Remove	-	0	-	-	-

Закінчення таблиці 4.12

4	Оптимізація	9	-	Remove	-	0	-	-	Adobe Photoshop Fix
5	Налаштування під користувача	6	Remove	Adobe Photoshop Fix	-	0	-	-	-
6	Приватність	19	Remove	-	-	0	-	-	Adobe Photoshop Fix
7	Технічна підтримка	14	Remove	-	-	0	Adobe Photoshop Fix	-	-

Таблиця 4.13. SWOT аналіз стартап-проекту

<p>Сильні сторони (S):</p> <p>Прагматичність системи через її легкість роботи;</p> <p>Простота у використанні;</p> <p>Зменшення затрат у редагуванні;</p> <p>Наявність відкритого вихідного коду;</p> <p>Збереження приватності інформації користувача.</p>	<p>Слабкі сторони (W):</p> <p>Неоптимізованість алгоритму;</p> <p>Швидкість роботи системи;</p> <p>Потрібне з'єднання з інтернетом</p>
<p>Можливості (O):</p> <p>Зворотній зв'язок з клієнтурою компанії для спроможності розвивати проект в інші напрямки.</p>	<p>Загрози (T):</p> <p>Автономна робота алгоритмів на електронному пристрої;</p> <p>Складність роботи алгоритму при невиявлених випадках використання додатку.</p>

Таблиця 4.14. Альтернативи ринкового впровадження стартап-проекту

№	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Безкоштовне надання певного функціоналу у користування споживачам на обмежений термін	Головний ресурс – люди, даний ресурс - наявний	2-3 місяці
2	Реклама	Залучення власних коштів для реклами товару	1-2 місяці
3	Написання статей та опис товару на відомих ресурсах	Головний ресурс – час, даний ресурс - наявний	2-3 тижні
4	Презентація товару на хакатонах й інших ІТ заходах	Ресурс – час та гроші для участі, наявні	1-3 місяці

#### 4.4. Розроблення ринкової стратегії проекту

Таблиця 4.15. Вибір цільових груп потенційних споживачів

№	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Туристи, які знаходяться в іноземній державі, де іноді важко знайти розетку та потрібно економити час для вивчення нового	Присутня	Середній	Майже відсутня	Легка
2	Звичайні користувачі які хочуть зменшити витрати для досягнення результату	Присутня	Середній	Присутня	Середня
Які цільові групи обрано: 1, 2					

Відповідно до проведеного аналізу можна зробити висновок, що підходящою цільовою групою для розповсюдження даного програмного

продукту є звичайні користувачі. Відповідно до стратегії охоплення ринку збуту товару обрано стратегію масового маркетингу.

Таблиця 4.16. Визначення базової стратегії розвитку

Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
Надання функціональності, що відсутня у товарів-замінників, підтримка клієнтів	Проведення реклами, освітлення унікальної функціональності через інтернет ресурси та інші канали, контакт напряду з споживачами;  формування лояльності і прихильності споживачів	Зниження ступеню замінності товару;  Прихильність клієнтів;  Відмітні властивості товару;  Відмітні характеристики товару;	Стратегія диференціації

Таблиця 4.17. Визначення базової стратегії конкурентної поведінки

Чи є проект «першопрохідцем» на ринку	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, які?	Стратегія конкурентної поведінки
Ні, оскільки є товари-замінники, але дані товари замінники не мають деякого необхідного функціоналу	Так, ціль компанії знайти нових споживачів та, частково, забрати існуючих у конкурентів задля задоволення потреб останніх	Компанія частково копіює характеристики товару конкурента, основна ціль компанії розробка нового унікального функціоналу, з підтримкою основного функціоналу конкурентів	Стратегія заняття конкурентної ніші

Таблиця 4.18. Визначення стратегії позиціонування

№	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту
1	Зручність	Диференціація	Спроможність економити час на редвгуванні фотографій	Спроможність вивчаючи важкі програми легко видалити рухомі об'єкти
2	Відкритість вихідного коду	Диференціація	Перспектива розвитку проекту	Розвиток в науці
3	Документація	Заняття конкурентної ніші	Можливість більше дізнатися про іноземні держави	Економія ресурсів

Відповідно до проведеного аналізу можна зробити висновок, що стартап-компанія вибирає як базову стратегію розвитку – стратегію диференціації, як базову стратегію конкурентної поведінки – стратегію заняття конкурентної ніші.

#### 4.5. Розроблення маркетингової програми стартап-проекту

Таблиця 4.19. Визначення ключових переваг концепції потенційного товару

№	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	В просторі в іноземній державі	Економія ресурсів	Швидкість, та легкість у використанні
2	В повсякденному житті	Економія ресурсів	Відсутність аналогів

Таблиця 4.20. Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
1. Товар за задумом	Видалення рухомих об'єктів на серії знімків		
2. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх/Тл/Е/Ор
	Зручність	Нм	Е
	Швидкість роботи	Нм	Тх
	Оптимізація	Нм	Тх
	Технічна підтримка	Нм	Тх
	Приватність	Нм	Тх
	Пакування: дані упаковані в файл формату CSV		
3. Товар із підкріпленням	До продажу: наявна повна документація, акції на придбання декількох ліцензій, знижки для певних сегментів на покупку товару		
	Після продажу: додаткова підтримка спеціалістів налаштування, підтримка з боку розробника		
За рахунок чого потенційний товар буде захищено від копіювання: захист інтелектуальної власності, патент			

В/Нв – відчутні/невідчутні; М/Нм – монотонні/немонотонні; Пр/Нпр – параметричні/непараметичні; Вр/Тх/Тл/Е/Ор – вартісні/ технічні/ технологічні/ ергономічні/ органолептичні; О/К/С – обов'язкові/ кількісні/ сюрпризні

Таблиця 4.21. Визначення меж встановлення ціни

Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
200 – 1000 грн./од.	1000 – 2000 грн./од.	6 000 – 15 000 грн./міс.	200-50 грн./од.

Таблиця 4.22. Формування системи збуту

Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
Всі користувачі будуть купляти товар по одинці	Можливість скачувати додаток влюбий час, у будь-якому місці.	2 рівня (посередник + клієнт)	Роздріб

Таблиця 4.23. Концепція маркетингових комунікацій

№	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання реklamного повідомлення	Концепція реklamного звернення
1	Туристи	Форуми, приватні зустрічі з компанією розробником	Дослідження	Показати можливість користування продукцією для клієнта	Рекламне звернення спрямоване до потенційних клієнтів, де показуються плюси користування системою
2	Звичайні користувачі		Фотографування	Показати можливість користування продукцією для клієнта	

Як результат було створено ринкову (маркетингову) програму, що включає в себе визначення ключових переваг концепції потенційного товару, опис моделі товару, визначення меж встановлення ціни, формування системи збуту та концепцію маркетингових комунікацій.

## Висновки по розділу

В четвертому розділі описано стратегії та підходи з розроблення стартап-проекту, визначено наявність попиту, динаміку та рентабельність роботи ринку, як висновок було вказано що існує можливість ринкової комерціалізації проекту. Розглянувши потенційні групи клієнтів, бар'єри входження, стан конкуренції та конкурентоспроможність проекту було встановлено що проект є перспективним. Розглянуто та вибрано альтернативу впровадження стартап-проекту та доведено доцільність подальшої імплементації проекту.



## ВИСНОВКИ

В першому розділі було проаналізовано існуючі рішення для видалення об'єктів з фотографії такі як : “Remove”, “Adobe Photoshop Fix”, “Zoner Photo Studio” Це справді дуже хороші додатки в якому є багато можливостей, і в ній теж є можливість прибрати різні предмети з фотографії, але їх потрібно прибрати вручну і в ній немає автоматизації виділення рухомих об'єктів та видалення їх. Якщо зайвий об'єкт розташований на не однотонному фоні, то ці методи не є ідеальними, при обробці великої зони фотографії цими способами буде помітно не озброєним оком недоліки зробленої роботи. Також для цього потрібен ПК але це вже не практично, на це потрібно витратити багато часу, та навички, отже потрібен такий додаток який зможе впоратися з цією задачею.

Нейронні мережі можуть дати нам певне уявлення про те як мозок працює, як інтегрує інформацію. Але справжнє питання полягає в тому, чому і як вся ця обробка в людині супроводжується досвідченим внутрішнім життям і чи може штучна нейронна мережа досягти такого рівня як біологічна нейронна мережа яка є в людині з народження?

В другому розділі ми розглянули різні нейронні мережі а саме вони нас змушують задуматися, чи можуть нейронні мережі стати пензлем для художників - новим способом створення, візуальні концепції, або, можливо, навіть пролити світло на коріння творчого процесу в цілому.

Загалом, нейронні мережі зробили комп'ютерні системи більш корисними, роблячи їх більш схожими до біологічної нейронної мережі людини. Таким чином, наступного разу, коли ви думаєте, що вам може сподобатися, то ваш мозок буде настільки ж надійним, як комп'ютер, подумайте ще раз будьте вдячними, що у вас є така чудова нейронна мережа, яка вже встановлена у вашій голові!

Протягом останнього десятиліття в області машинного навчання домінують так звані глибокі нейронні мережі, які користуються перевагами покращення в обчисленні потужності та доступності даних. Підтип нейронної

мережі має назву згорткова нейронна мережа, добре підходить для завдань пов'язаних з зображенням. Мережа навчається шукати різні функції, такі як краї, кути та кольорові відмінності, та об'єднати їх у більш складні фігури. Для виявлення об'єкта система має оцінювати розташування імовірних об'єктів та їх класифікувати.

В третьому розділі ми на практиці, ми створили робочу програму MATLAB Fast R-CNN. Ми дізналися, що найбільш складною частиною впровадження глибокої системи навчання є збір даних для навчання та проведення самого навчання. Доступні набори даних є корисною відправною точкою як для дослідження, так і для практичних реалій. Тривалість навчання може бути ще скорочена за допомогою попередньо навченої мережі. Навіть якщо кінцева система не має тих же класів об'єктів, що і еталонні дані, візуальні проблеми є достатньо універсальними, аби отримати вигоду від детекторів, навчених для іншої проблеми. Оптимальні нижні шари згорткової мережі часто схожі, незалежно від проблеми, так само, як людське око використовує ті ж поля для всіх візуальних завдань. Таким чином, має сенс ініціалізувати шари за допомогою попередньо навченої мережі.

Що стосується реалізації, що не існує простих "нестандартних" рішень для ефективного впровадження згорткових мереж. Поточні програмні засоби, такі як Caffe та MatConvNet, вимагають спеціальних навичок. Якщо можна використовувати такі інструменти, створення робочої реалізації не є надто складним. Тим не менше, інструменти досить вигадливі щодо сумісності версій програмного забезпечення та апаратного забезпечення. Що стосується android додатку був розроблений простий в користуванні інтерфейс який об'єднує людей з такою неймовірною системою як нейронна мережа, в якій волицезний потенціал в різних сферах.

В четвертому розділі описано стратегії та підходи з розроблення стартап-проекту, визначено наявність попиту, динаміку та рентабельність роботи ринку, як висновок було вказано що існує можливість ринкової комерціалізації

проекту. Розглянувши потенційні групи клієнтів, бар'єри входження, стан конкуренції та конкурентоспроможність проекту було встановлено що проект є перспективним. Розглянуто та вибрано альтернативу впровадження стартап-проекту та доведено доцільність подальшої імплементації проекту.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Саймон Хайкин. Нейронные сети: Полный курс. Киев. 2008. 1103 с.
2. Paul Viola. Detecting Pedestrians Using Patterns of Motion and Appearance. 2005. Vol. 63(2). P. 153-161. DOI: 10.1007/s11263-005-6644-8.
3. Evgeniy Bart, Evgeny Byvatov and Shimon Ullman. View Invariant Recognition Using Corresponding Object Fragments. 2004. vol. 3022. P. 152-165.
4. Liu Y. and Goto S. An efficient and accurate approach of circular object detection in color images. 2014. vol. 40. P. 26–36.
5. Уоссермен Ф. Нейрокомпьютерная техника: Теория и практика. Мир. 1992. 240 с.
6. Agoston M.K. Computer graphics and geometric modeling: Implementation and algorithms. 2005. P. 290-306. DOI: 10.1007/b138805.
7. Amosov O.S. High-speed neurofuzzy algorithms for filtering the mobile object trajectory parameters. 2016. P. 389-392.
8. Jang-Shing Roger Jang. ANFIS: adaptive-network-based fuzzy inference system. 1993. Vol. 23(3). P. 665-685. DOI: 10.1109/21.256541.
9. Ouyang W. Joint Deep Learning for Pedestrian Detection. 2013. P. 2046-2063. DOI: 10.1109/ICCV.2013.257.
10. Enzweiler M. Monocular Pedestrian Detection: Survey and Experiments. 2009. Vol. 31(12). P. 2169-2195. DOI: 10.1109/TPAMI.2008.260.
11. Dalal N., and Triggs B. Histograms of oriented gradients for human detection. In Computer Vision and Pattern Recognition. 2005. vol. 1, P. 876–893.
12. Felzenszwalb P., Felzenszwalb F. and Huttenlocher D., Huttenlocher P. Efficient graphbased image segmentation. International journal of computer vision 59. 2004. P. 167–181.
13. Hoiem D., Efros A., Efros A., and Hebert, M. Putting objects in perspective. International Journal of Computer Vision 80. 2008. P. 3–35.

14. Krizhevsky A., Sutskever I. and Hinton G. E. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems. 2012. P.1047–1105.
15. Плохута Д.О., Корнага Я.І. Система виділення людських силуетів, що рухаються, на серії зображень за допомогою нейронної мережі. Вчені записки Таврійського Національного Університету Імені В.І. Вернадського. Том 29(68) №5 2018 частина 2. С. 31-35.

## ДОДАТКИ

## ДОДАТОК А

UML діаграма класів камери

## ДОДАТОК Б

UML діаграма класів менеджера фотографій



## ДОДАТОК В

### Схема роботи нейронної мережі

## ДОДАТОК Г

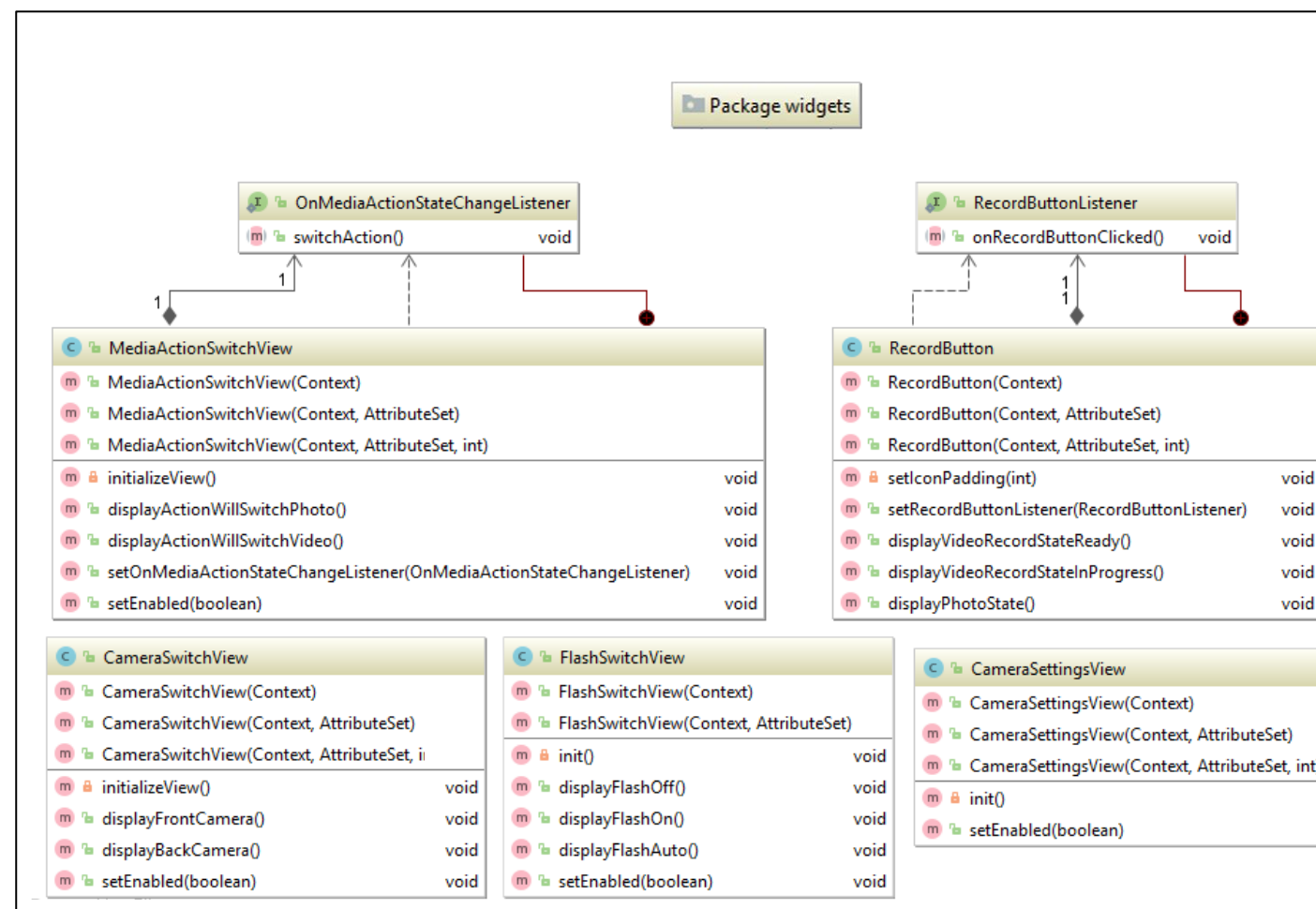
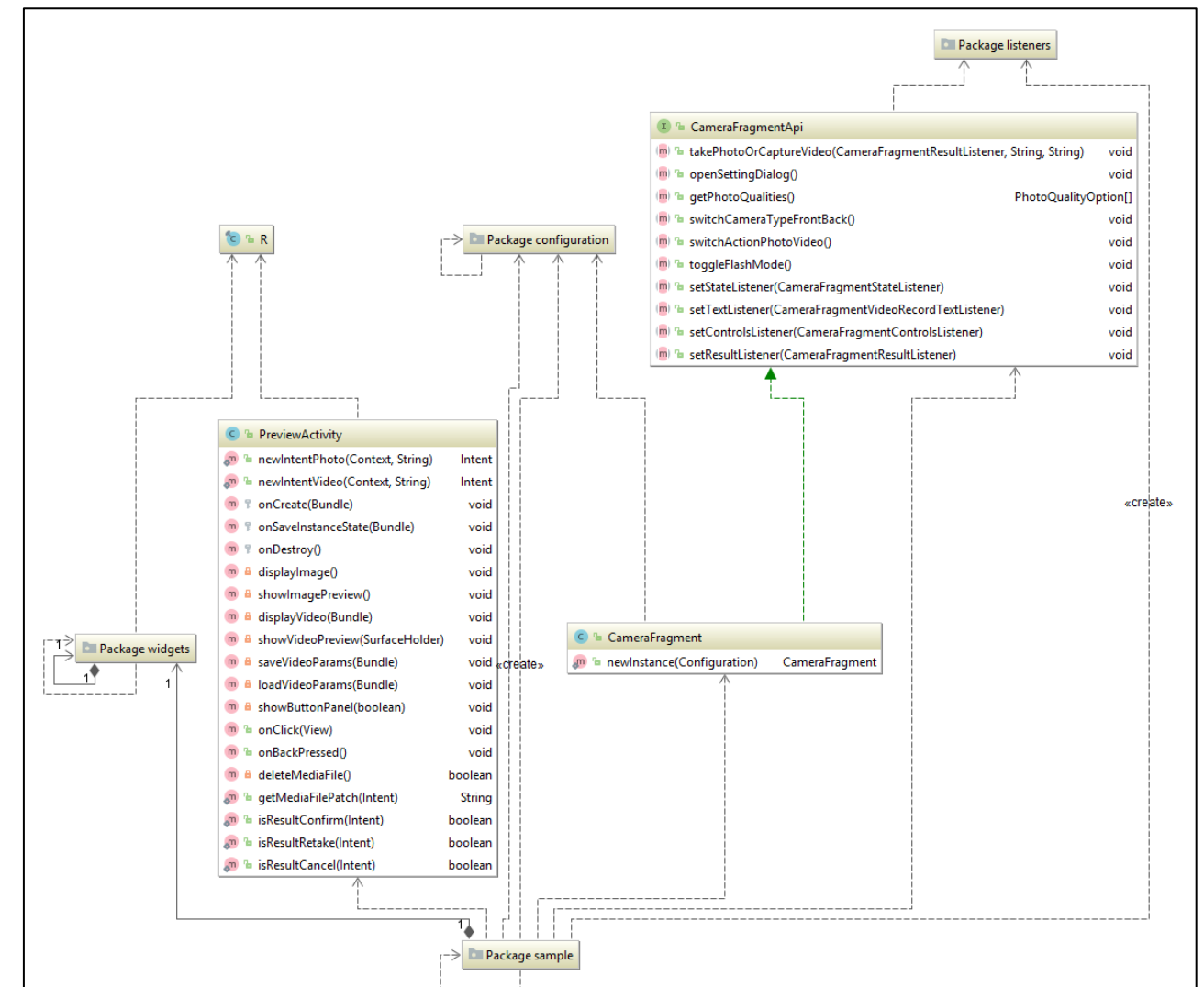
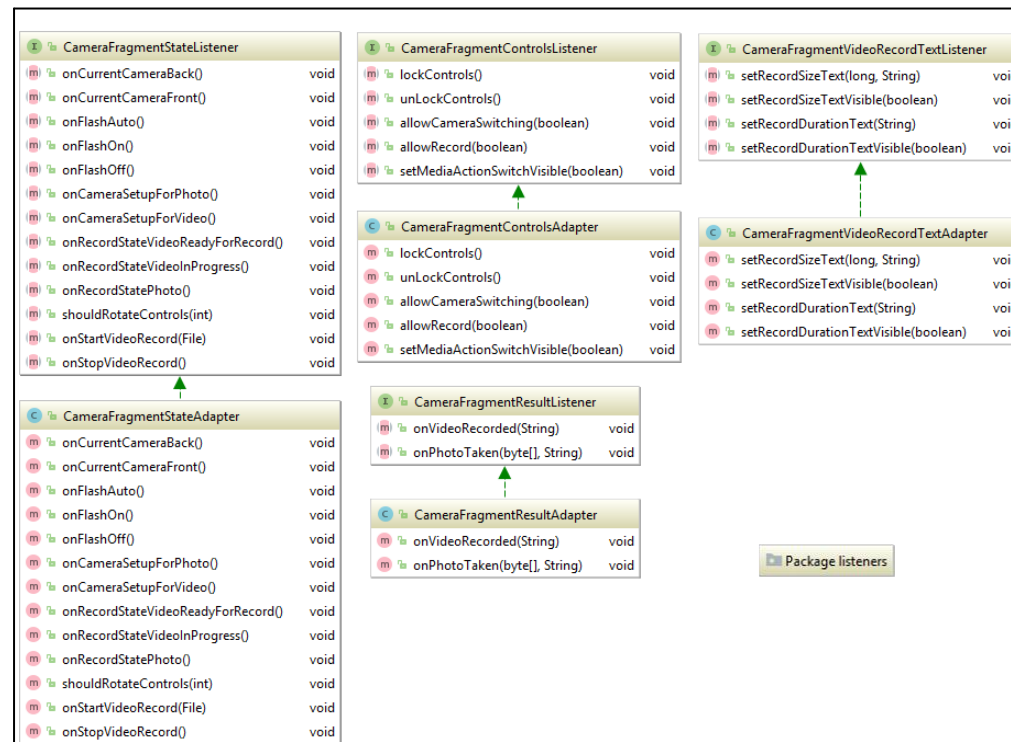
### База даних

ДОДАТОК Д  
Інтерфейс додатку

ДОДАТОК Е

Результати експериментів

# Схема класів камери

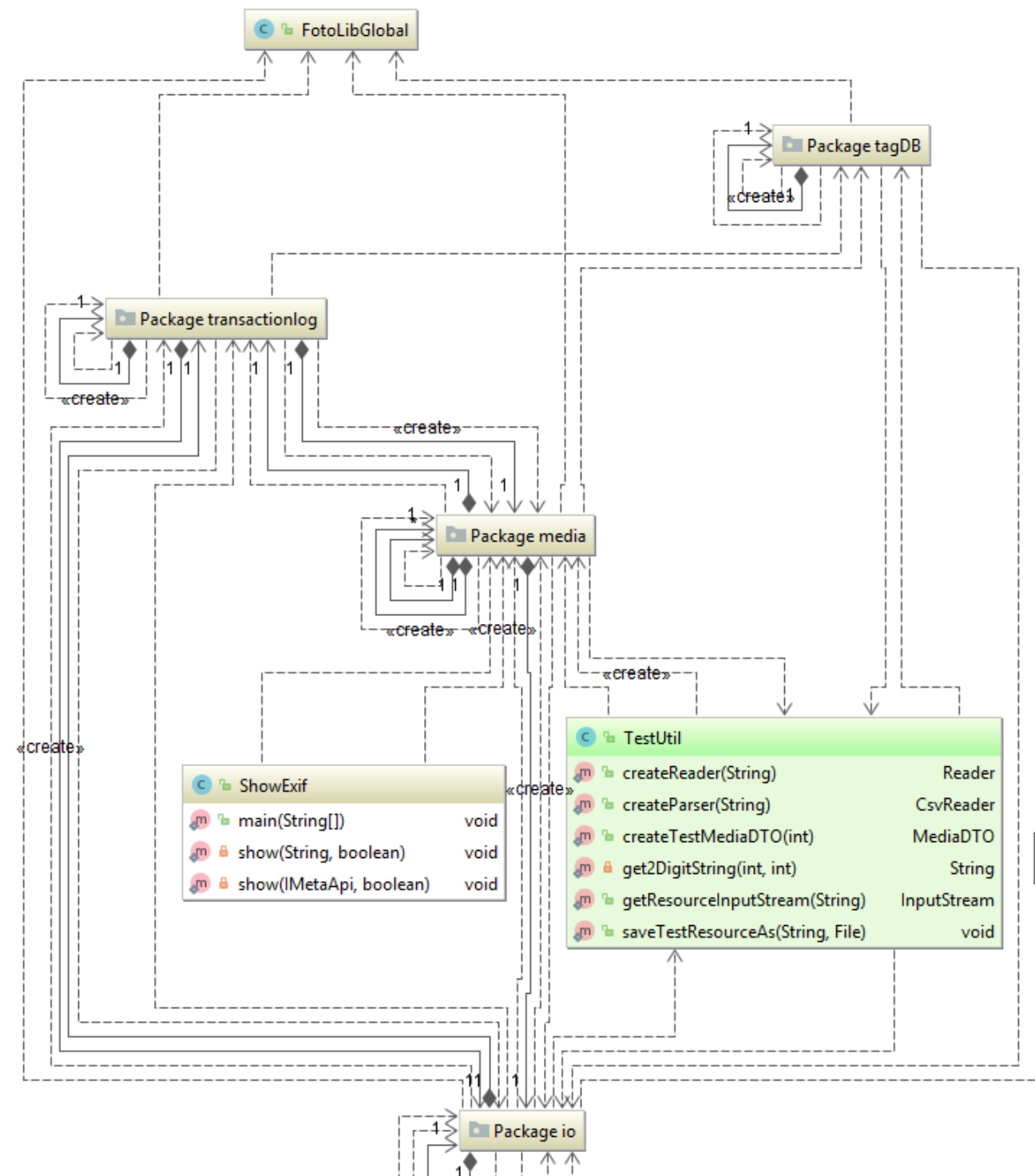


Демонстраційний плакат №\_\_  
до магістерської дисертації на тему  
„Система виділення динамічних об'єктів на серії зображень за  
допомогою нейронної мережі”

Розробив: \_\_\_\_\_

Прийняв: \_\_\_\_\_

# Схема класів менеджера фотографій

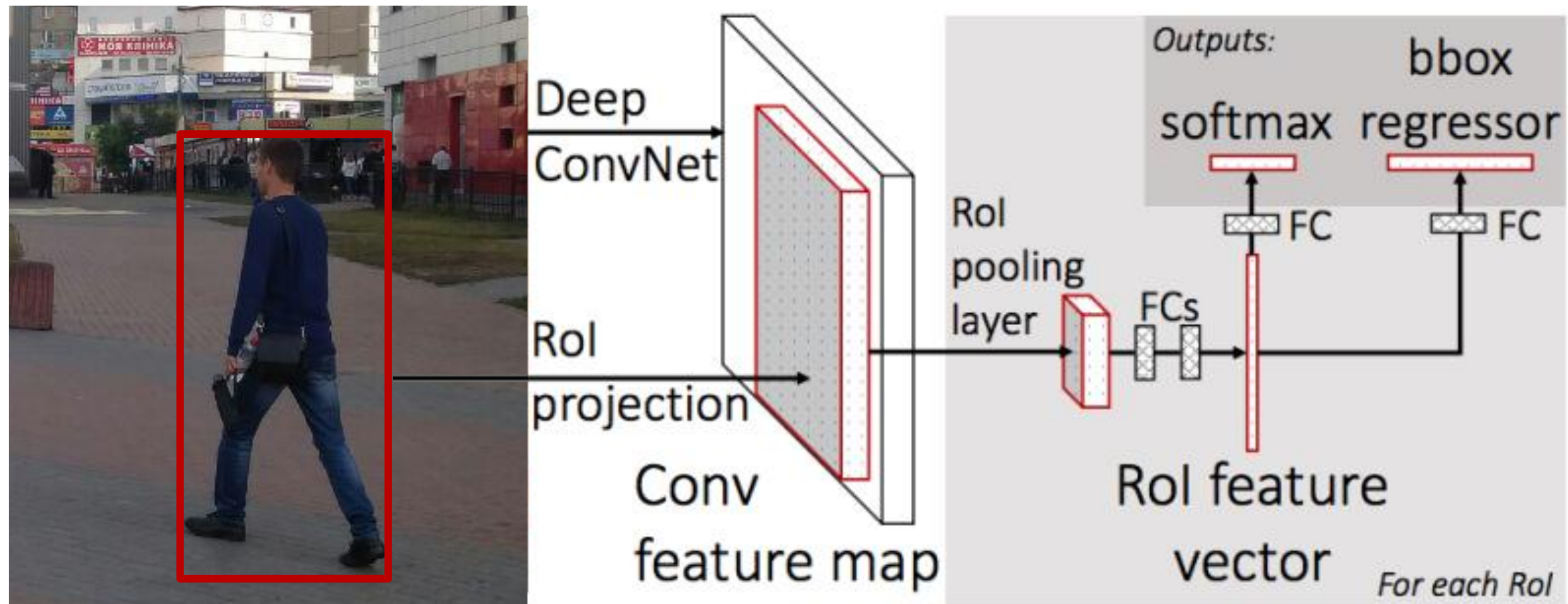


Демонстраційний плакат №\_\_  
до магістерської дисертації на тему  
„Система виділення динамічних об'єктів на серії зображень за  
допомогою нейронної мережі”

Розробив: \_\_\_\_\_

Прийняв: \_\_\_\_\_

# Схема роботи нейронної мережі

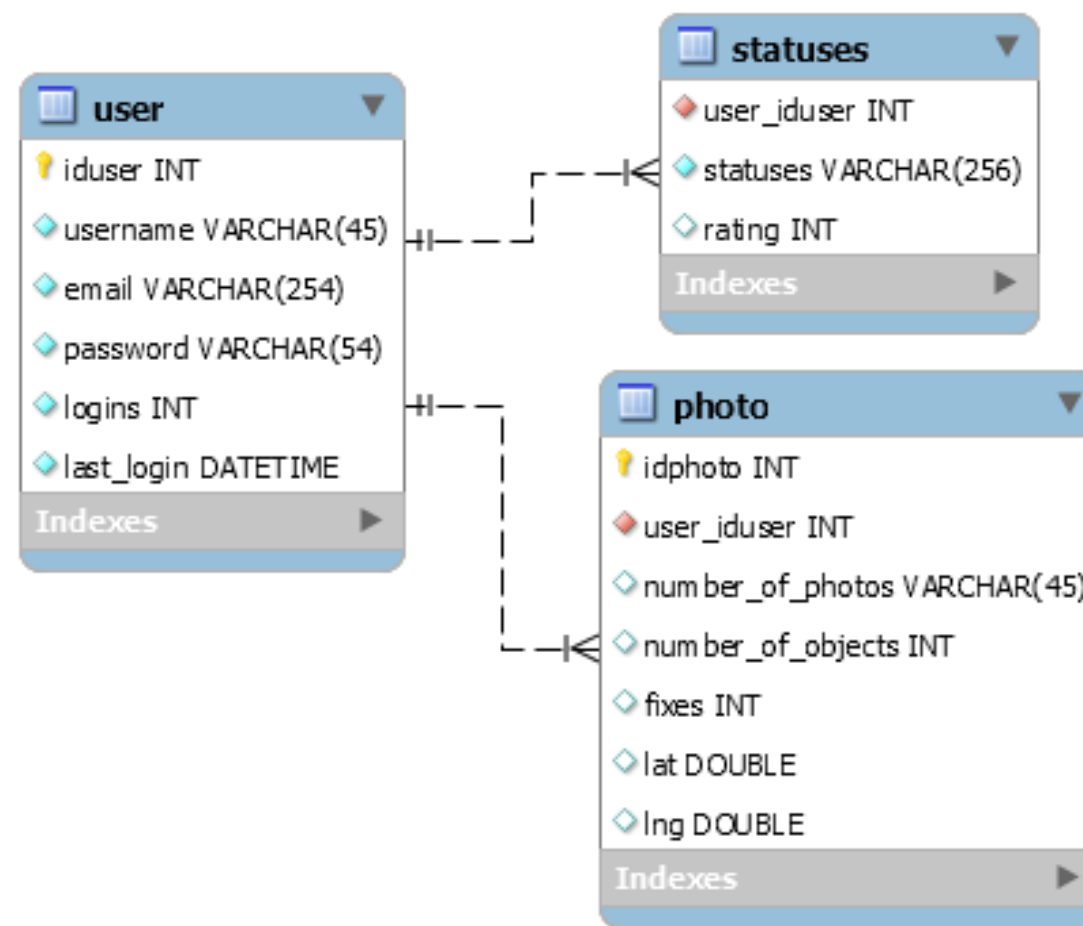


Демонстраційний плакат №\_\_  
до магістерської дисертації на тему  
„Система виділення динамічних об'єктів на серії зображень за  
допомогою нейронної мережі”

Розробив: \_\_\_\_\_

Прийняв: \_\_\_\_\_

# База даних



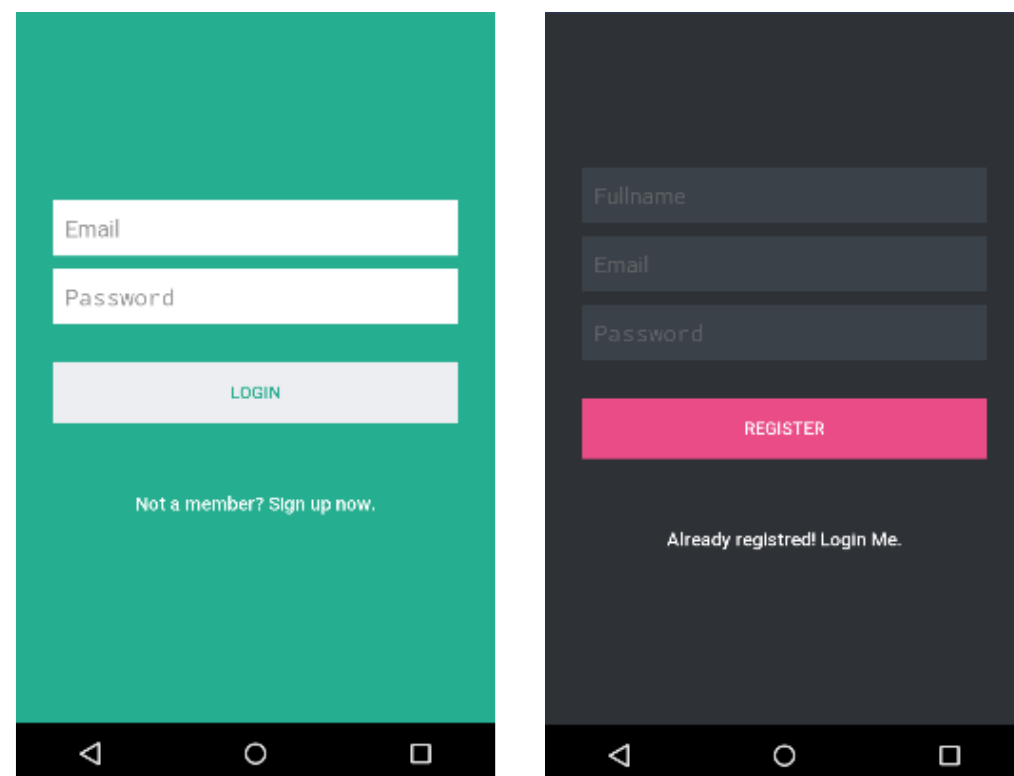
Демонстраційний плакат №\_\_  
до магістерської дисертації на тему  
„Система виділення динамічних об'єктів на серії зображень за  
допомогою нейронної мережі”

Розробив: \_\_\_\_\_

Прийняв: \_\_\_\_\_



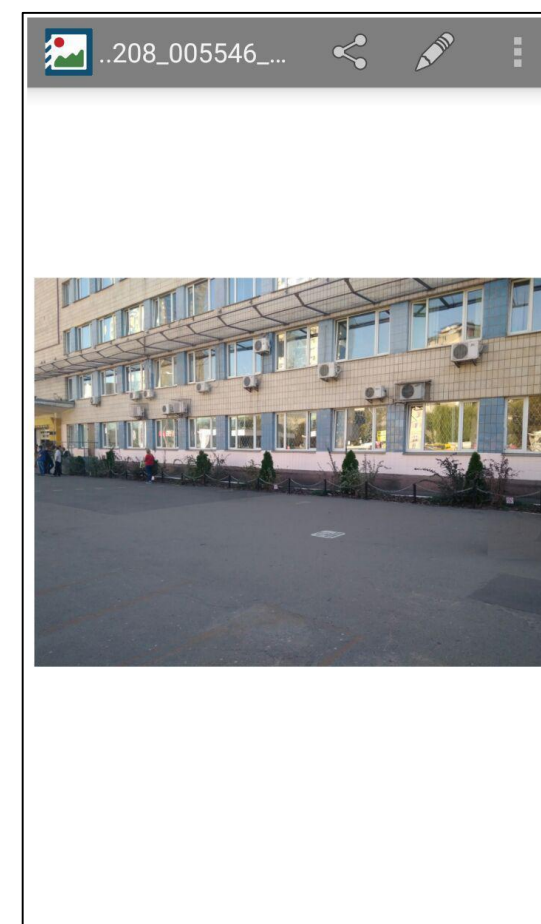
# Інтерфейс додатку



Вікно авторизації та реєстрації користувачів



Вікно активної камери



Вікно отриманого результату



Вікно історії здійснених фотографій

Демонстраційний плакат №\_\_  
до магістерської дисертації на тему  
„Система виділення динамічних об'єктів на серії зображень за  
допомогою нейронної мережі”

Розробив: \_\_\_\_\_

Прийняв: \_\_\_\_\_

# Результати експериментів



Фотографії до проходження навчання нейронної мережі



Результати після обробки фотографії



Виявлення об'єктів після проходження навчання нейронної мережі

Демонстраційний плакат №\_\_  
до магістерської дисертації на тему  
„Система виділення динамічних об'єктів на серії зображень за  
допомогою нейронної мережі”

Розробив: \_\_\_\_\_

Прийняв: \_\_\_\_\_